

**Oracle® Communications Session
Element Manager**

Configuration Guide

Release 7.3

Formerly Net-Net Central

October 2013

Copyright ©2013, 2012 Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

About this Guide	li
Overview	li
Configuring Different Versions of Net-Net SBCs	li
Configuration Timestamp	li
Revision History	lii
1 Net-Net System Basics	59
Introduction	59
Realms	59
What Is a Realm?	59
Nested Realms	59
Session Agents and Session Agent Groups	60
What Is a Session Agent?	60
Why You Need Session Agents	60
How to Use Session Agents	60
What is a Session Agent Group?	60
High Availability (HA)	61
2 Getting Started	63
Introduction	63
Overview	63
In this Chapter	63
Net-Net Central Applications	63
Managed Devices	64
Configuration Elements in this Guide	64
Net-Net Central Installation Requirements	66
System Requirements	66
Information You Need	67
Net-Net Central Components Required	68
Opening Ports on the Firewall	68
Net-Net Central Virtual Environment on Linux	69

Before You Start	69
Installing VM NNC.....	70
Before a New Installation	71
Configuring Linux.....	71
OpenSSL Required for HTTPS	72
Creating nncentral Group and User	73
Defining sudo Privileges	73
Installing Net-Net Central	76
Installing Net-Net Central.....	76
Running Setup	76
Typical Setup.....	78
Custom Setup	85
Accessing Custom Options	85
Cluster Management	86
Manage Cluster Members	87
Route Management Central.....	92
Mail Server	94
Starting the NNC Server	98
Stopping the NNC Server	98
Migrating Data to Net-Net Central	99
Net-Net EMS 6.x.....	99
Net-Net Route Manager Central 1.x.....	99
Requirements	99
Before You Migrate Data	100
Data Migration from Net-Net EMS 6.x	100
Migrating from Net-Net Route Manager Central 1.x	102
Running Database Migration Again.....	103
Data Migration Logging.....	104
Error Logging	104
Deploying Migrated Data on Net-Net Central Cluster.....	105
Mapping Device Groups to User Groups.....	105
Backup and Restore Database Servers.....	107
Backing Up with Server Shutdown.....	107
Backing Up with Server Running	107
Restoring Database Backups	107
Configuring the Net-Net SBC for Net-Net Central Interaction	108
Configuring the SNMP Interface.....	108
Starting the Net-Net Central Client and Connecting to the Server.....	110
Verifying the Client System Settings.....	110
Starting the Net-Net Central Client	111

Net-Net SBC License Information	111
Accessing License Information	113
Unlicensed Net-Net Systems	115
Obtaining a License	115
Loading a Configuration	116
Associating Devices	116
Loading a Local Configuration Copy	117
After your Configuration is Loaded	118
Performing Actions with Loaded Configurations	118
About Devices Data	119
Removing an Associated Device	120
Relationship with ACLI	120
Working with Configuration Views	121
Navigating Between Configuration Views	121
Using the Default View	122
Using the ACLI View	123
Using the List View	124
User Interaction	125
Methods	125
Error Messages	129
Using Tool Tips	129
Configuring Single-Instance Configuration Elements	130
Modifying Primary Key Attributes	130
Configuration Timestamp	130
Saving and Activating Configurations	130
Provisioning a Device in the LCC	131
Viewing Modifications in the LCV	133
Suggested Methods	135
Applying Provisioning Changes in Net-Net Central	137
Saving and Activating Configuration	137
Locking Devices	141
Setting Up NTP Synchronization	143
RADIUS Authentication	144
How It Works	144
PAP Handshake	146
CHAP Handshake	146
MS-CHAP-v2 Handshake	147
Management Protocol Behavior	148
Configuring RADIUS Authentication	149
SSH RADIUS Authentication VSA Support	153

3	System Configuration	155
	Introduction	155
	General System Information	155
	Configuring General System Information	156
	Configurable Timeout for Debug Logging	157
	Configuring Timeout For Debug Logging	157
	Physical Interfaces	158
	Overview	158
	Types of Physical Interfaces	158
	Before You Configure	159
	Configuring Physical Interfaces	160
	Configuring Physical Interfaces	160
	Interface Utilization: Graceful Call Control, Monitoring, and Fault Management	162
	Calculation Overview	162
	Alarms	162
	Configuring Utilization Thresholds for Media Interfaces	163
	Configuring Graceful Call Control	164
	SNMP Support	164
	HDR Statistics	165
	Network Interfaces	166
	Overview	166
	IP Configuration	166
	Configuring Network Interfaces	167
	Configuring Multiple ICMP Addresses	170
	SNMP	172
	Overview	172
	Configuring SNMP	173
	SNMP Configuration Overview	173
	Configuring SNMP	173
	Alarm Thresholds	177
	Configuring Alarm Thresholds	177
	ETC Monitoring	178
	ETC CPU and Memory Monitoring	179
	Configuration Instructions	179
	Syslog and Process Logs	180
	Overview	180
	Configuring Syslog and Process Logs	180
	Message Rate Statistics	185

Host Routes	187
Overview	187
Configuring Host Routes	188
Holidays	189
Configuring Holidays	189
Enhanced Control of UDP and TCP Ports	190
ACLI Instructions and Examples	190
HDR Trap for Data Push Success	192
Configuring HDR Data Push Success Trap	192
DNS Transaction Timeout	193
Retransmission Logic	193
Configuring DNS Transaction Timeout	193
Persistent Protocol Tracing	194
About Persistent Protocol Tracing	194
About the Logs	195
Configuring Persistent Protocol Tracing	195
Accounting	196
SIP Session Accounting	196
Accounting Configuration	197
Packet Trace	200
How It Works	200
Packet Trace Scenarios	201
Configuring Packet Tracing	203
Alarm Synchronization	204
Caveats	205
Configuring Alarm Synchronization	205
Configuring TCP Timers	206
Configuring TCP Timers	206
SIP over SCTP	211
SCTP Terminology	211
SCTP Overview and Comparisons	211
How Is SCTP Different from TCP?	212
How Is SCTP Different from UDP?	212
Configuring SCTP Delivery Mode	212
Configuring SCTP for Realms	213
Configuring SCTP for Session Agents	214
IPv6 Support	215
Licensing	215

Globally Enabling IPv6	215
IPv6 Address Configuration	216
IPv6 Default Gateway	219
Network Interfaces and IPv6	219
Access Control List Support	219
DNS Support	220
Homogeneous Realms	220
Compact Flash Memory Use With Local CDRs / FTP Push	220
Compact Flash Memory Specifications	220
Compact Flash Standby Replication	220
Local CDR Storage Directory	221
FTP Push Backup	221
Local CDR File Compression	221
CDR Decompression	221
Writing Local CDRs to a Compact Flash Module	221
Compact Flash Module Monitoring	222
Multiple CDR Push Receivers	224
RADIUS Account Server Prioritization	228
Server Prioritization Example	228
Setting Priority for Account Servers	229
NIU GigE Utilization	229
Segmenting Traffic	230
SNMPv3 Secure Traps	231
Authentication and Privacy	231
Enabling SNMPv3	232
Consideration for HA Nodes	232
Protection Against DDoS Attack From Behind a NAT	236
Configuring Protection Against DDoS Attacks From Behind a NAT	236
CSV Filename Sequence Numbering	238
Adding Filename Sequence Numbers	239
FTP/SFTP Server Upload Verification	239
Server Requirements	240
Configure Upload Verification	240
Downgrade Warning	240
Throttled Call Teardown to/from Session Agent	241
Support for Temporary Public User Identities Including Multi-SIM Scenarios	242
NN3800 Licensing Changes	242
NN3820 Licensing Changes	242
Network Management Control State	242

SNMP Support	242
Increase Session Limit and Remove 10X Reg Limit.....	245
Surrogate Agent Refresh on Invalidate	245
IPv6 SRTP MIKEY Key Exchange	246
Suppress DUP 200	246
Fragmented IP Packet Policing	246
Enabling Fragmented IP Packet Policing	246
Option Hairpin Released Flows.....	247
Hairpin Released Flows Option.....	247
Configurable MTU Size for IPv4 and IPv6.....	248
Configuring MTU Size for IPv4 and IPv6	248
P-CSCF PANI Enhancements	249
SNMP Extras	249
Asymmetric Payload Type	250
Configuring Asymmetric Payload Types	250
Dual Stack Support.....	250
Media Stop Time VSA in CDRs.....	251
Media Stop Time VSAs.....	251
Media Stop Time Calculation.....	251
HA Caveat	251
Diameter Rf Charging Buffering and Storage.....	252
About Buffering.....	252
About Storage	252
Configuring Diameter Rf Buffering and Storage	252
Diameter Attribute ID Range Selection	253
Configuring Diameter Rf Attribute ID Range.....	253
Session Plug-In Language (SPL)	254
Enabling SPL Plugins	254
Uploading SPL Plugins.....	254
Configuring SPL Plugins	255
SPL Parameter Configuration	256
Executing SPL Files.....	256
Old File Remover	258
Specifics and Caveats	258
Configuring Old File Purging.....	259
Temporary File Naming for an Open CDR File	260
Operational Details.....	260
HA Considerations	261

Caveats.	261
Disabling Temporary CDR File Naming.	262
4 Realms and Nested Realms	263
Introduction.	263
Overview	263
Configuring Realms.	265
Before You Configure	265
Configuration Overview.	265
Configuration Timestamp	267
Configuring Realms	267
RTP-RTCP UDP Checksum Generation	270
Configuring UDP Checksum Generation.	270
Nested Realms	270
Configuring Nested Realms.	272
Configuring Nested Realms.	272
Aggregate Session Constraints: Nested Realms	276
Impact to Other Session Constraints and Emergency Calls	276
Configuring Session Constraints for Nested Realms.	277
Aggregate Session Constraints Per Realm	278
Configuring Session Constraints per Realm.	278
Realm-Based Packet Marking	279
About TOS/DiffServ	279
Packet Marking for Media	280
Configuring Packet Marking by Media Type	280
Configuring Packet Marking for Signaling.	282
SIP-SDP DCSP Marking/ToS Bit Manipulation.	284
Configuring TOS Bit Manipulation.	285
Steering Pools	286
Configuration Overview.	286
Configuring Steering Pools	286
Multiple Interface Realms	288
How It Works	288
Steering Pool Port Allocation	290
Configuring Multiple Interface Realms.	290
Media over TCP	291
TCP Bearer Conditions.	291
TCP Port Selection	292

Configuring Media Over TCP	296
Configuring a Preferred Codec	297
How It Works	297
Configuring a Preferred Codec.	298
Restricted Media Latching	299
About Latching	299
How it Works.	300
Configuring Restricted Media Latching	302
Media Release Across SIP Network Interfaces.....	303
Example	303
Configuring Media Release Across SIP Network Interfaces	304
Media Management Behind the Same IP	304
Additional Media Management Options	304
Configuring Media Management Behind the Same IP	305
Media Release Behind the Same NAT IP Address.....	306
Configuring Media Release Behind the Same NAT IP Address	306
Per Realm Accounting Control	308
SIP Per-User CAC.....	309
How It Works	309
Configuring SIP Per-User CAC	310
Bandwidth CAC for Media Release	311
Configuring Bandwidth CAC for Media Release	312
Media Profiles Per Realm	312
Call Admission Control and Policing	312
Configuring Multiple Media Profiles Per Realm.....	313
Customizing Media Profiles	314
Multiple Media Profiles.....	315
Use Case 1	315
Use Case 2	315
Configuring a Subname for a Media Profile	316
STUN Server	317
About STUN Messaging.....	317
STUN Server Functions on the Net-Net SBC.....	318
RFC 3489 Procedures	318
Configuring a STUN Server	319
Multi-System Selective SRTP Pass-Through	321
Configuring Multi-System Selective SRTP Pass-Through	321
IP Identification (ID) Field.....	321

5	SIP Signaling Services	323
	Introduction	323
	About the Net-Net SBC and SIP	323
	Types of SIP Devices	323
	Basic Service Models	324
	About SIP Interfaces	325
	SIP INVITE Message Processing	326
	Configuring the Net-Net SBC for SIP Signaling	327
	Using the Net-Net Central to Configure the Net-Net SBC	327
	Configuration Timestamp	328
	Home Realm	328
	Overview	328
	Configuring the Home Realm	329
	SIP Interface	331
	Overview	331
	About SIP Ports	331
	Proxy Mode	331
	Redirect Action	332
	SIP MADDR Resolution	332
	Configuring SAG-Based MADDR Resolution	333
	Trust Mode	334
	Configuring a SIP interface	335
	Timers	341
	Overview	341
	Configuring SIP Timers	342
	SIP Options Tag Handling	344
	Overview	344
	Configuration Overview	344
	Configuring SIP Options Tag Handling	345
	SIP Options	347
	Overview	347
	Global SIP Options	347
	SIP Interface Options	352
	SIP Session Agent Options	354
	SIP Realm Options	354
	Configuring SIP Options	354
	Security	360
	Denial of Service Protection	360
	Configuration Overview	361

Configuring Security	361
SIP Unauthorized Endpoint Call Routing	362
Configuring SIP Unauthorized Endpoint Call Routing	362
SIP NAT Function	363
Overview	363
About Headers	365
SIP NAT Function Cookies	366
Configuration Overview	368
Configuring SIP NAT Function	370
Realm Bridging	373
About SIP NAT Bridging	374
SIP NAT Bridge Configuration Scenarios	374
SIP NAT Bridge Configuration	376
Shared Session Agent	378
Hosted NAT Traversal	378
About SIP HNT	378
Working with Multiple Domains	382
HNT Configuration Overview	383
Configuring SIP Hosted NAT Traversal	384
Adaptive HNT	388
Overview	388
Adaptive HNT Example	389
Server Redundancy	390
Overview	390
Configuration Overview	391
Configuring SIP Server Redundancy	392
Distributed Media Release	392
Overview	392
Overview of SIP DMR Configuration	394
Configuring Distributed Media Release	395
Configuring the Realm Configuration	397
Add-On Conferencing	398
Overview	398
Configuring Add-On Conferencing	400
Roaming	402
Overview	402
Process Overview	402
Configuring SIP Roaming	404
Embedded Header Support	406

Configuring Embedded Header Support	406
Header and Parameter Manipulation	407
Header Manipulation Rules	407
Header Element Rules	407
About SIP Header and Parameter Manipulation	407
Configuring Header and Parameter Manipulation	409
Dialog-Matching Header Manipulation	415
About Dialog-Matching Header Manipulations	415
Enabling Dialog- Matching Header Manipulation	418
Configuring Built-In SIP Manipulation	418
Unique Header Manipulation Rule Regex Patterns	421
Manipulation Pattern Per Remote Entity	421
Reject Action	421
Log Action	423
Changes to Storing Pattern Rule Values	423
Name Restrictions for Manipulation Rules	423
New Value Restrictions	424
Dialog Transparency	424
Overview	424
Configuring Dialog Transparency	425
Route Header Removal	425
Configuring Route Header Removal	426
Via Transparency	427
How it Works	427
Configuring Via Transparency	428
Symmetric Latching	429
Configuring Symmetric Latching	429
Configuring RTCP Symmetric Latching	429
Number Normalization	430
How it Works	430
Configuring Number Normalization	431
Ensuring Placement of RADIUS Attributes in CSV File	433
Port Mapping	446
About SIP Port Mapping	446
How SIP Port Mapping Works	447
About NAT Table ACL Entries	447
Configuring SIP Port Mapping	449
SIP Port Mapping for TCP and TLS	450
Configuring TCP/TLS Port Mapping	451

SIP Registration Via Proxy	452
Considerations for Reg-Via-Key and Port Mapping	453
Request Routing	453
Configuring SIP Registration Via Proxy	453
Configurable Route Recursion.....	454
Example 1.....	455
Example 2.....	455
Configuring Route Recursion.....	456
Event Package Interoperability	458
Configuring Event Package Interoperability.....	459
REGISTER Forwarding After Call-ID Change	459
Configuring REGISTER Forwarding After Call-ID Change	460
Local Response Code Mapping	460
Configuring Local Response Code Mapping	461
Session Agent Ping Message Formatting.....	463
Configuring Session Agent Ping Message Formatting	463
PAI Stripping.....	464
SIP PAI Stripping Configuration	466
Configuring SIP PAI Stripping.....	466
SIP Statuses to Q.850 Reasons Mapping	468
SIP to SIP Calls	468
Configuring SIP Statuses to Q.850 Reasons Mapping.....	469
Calls Requiring IWF	470
Configuring Calls Requiring IWF.....	473
Trunk Group URIs	476
Terminology.....	476
Trunk Group URI Parameters	476
How It Works	480
Before You Configure	481
Setting the Trunk Group URI Mode for Routing	482
Configuring a Session Agent for Trunk Group URIs.....	482
Configuring a Session Agent Group for Trunk Group URIs.....	484
Setting a Trunk Group Context in a Realm.....	485
Trunk Group URIs with SIP Interface Registration Caching.....	485
Emergency Session Handling	486
Configuring Emergency Session Handling.....	487
Fraud Prevention	488
Configuring Fraud Prevention	488
Early Media Suppression	489

How it Works	490
About the Early Media Suppression Rule	491
Configuring Early Media Suppression	492
Early Media Blocking Enhancements	493
How It Works	494
Configuring Realm Groups	494
SDP-Response Early Media Suppression	496
How it Works for SIP-Based Addressing	496
How it Works with SDP-Based Addressing	496
Configuring SDP-Response Early Media Suppression	498
Configuring SDP-Response Early Media Suppression	499
SDP Anonymization	501
Configuring SDP Anonymization	501
Restricted Media Latching	502
About Latching	502
How it Works	502
Configuring Restricted Latching	504
Enhanced SIP Port Mapping	504
Anonymous Requests	504
Dynamic Transport Protocol Change	505
Privacy Extensions	507
How it Works	507
Privacy Types Supported	507
Examples	508
Configuring SIP Privacy Extensions	509
Configuring SIP Privacy Extensions	509
SIP Registration Cache Limiting	510
How It Works	510
About Registration Cache Additions, Modifications, and Removals	510
Registration Cache Alarm Threshold	511
Notes on Surrogate Registration	512
Configuring SIP Registration Cache Limiting	512
Configuring SIP Registration Cache Limiting after Retry	513
Configuring Registration Cache Alarm Threshold	514
Semi-Active IP Call Recording	515
How It Works	515
Adding a CRS	516
Configuring a CRS	517
Applying a Call Recording Server Entry to a Realm	518
Increased CRS Capacity	519

Globally Unique Call ID	519
X-UCID Notes	520
Configuring the UCID in SIP Messages	520
Transport Selection	521
Configuring SIP Transport Selection	522
TCP Connection Reuse	523
How It Works	523
Configuring SIP TCP Connection Reuse	523
Enforcement Profile and Allowed Methods	524
Configuring Enforcement Profiles and Allowed Methods	524
Applying SIP Enforcement Profiles	526
Local Policy Session Agent Matching	527
How It Works	528
Configuring Local Policy Session Agent Mapping	532
Implicit Service Routing	534
Configuring Implicit Service Routing	534
Contact Header Contact Formatting	535
Configuring Contact Header Contact Formatting	536
403 Response Code Override	537
Configuring 403 Response Code Override	537
SIPconnect	537
Configuring SIPConnect Support	538
Configuring SIPConnect	539
SIP Port Mapping Enhancement	542
How It Works	542
Configuring SIPconnect Support	543
HNT Contact Endpoint	545
Configuring HNT Contact Endpoint	545
Delayed Media Update	546
Previous Behavior	546
New Behavior	546
Configuring Delayed Media Update	547
Interface Response Code Mapping	547
Configuring SIP Interfaces	547
Port Mapping Based on IP Address	549
How It Works	549
Configuring Port Mapping	549

Registration Overload Protection	551
How It Works	551
Configuring SIP Registration Overload Protection	552
SIP Request Method Throttling	553
How It Works	553
Configuring Rate Constraints for Session Constraints	554
Configuring Rate Constraints for SIP Session Agents.	556
SIP Registration Response Mapping	557
Configuring SIP Registration Response Mapping: SIP Interface	558
Configuring SIP Registration Response Mapping: Local Response Map	559
Configurable Timers and Counters	560
Configuring Timers and Counters.	560
Mid-Call Re-Registration Following Service Route	561
Configuring SIP Mid-Call Re-Registration.	562
Random Registered-Contact	563
Configuring Random Registered-Contact	563
Suppress SIP Reliable Response Support for IWF, Phase 1	564
Configuring Reliable Response Support for IWF.	565
Response Blocking	566
Configuring a SIP Interface	566
Configuring Session Agents.	567
SDP Sendrecv Insertion	569
Configuring SDP sendrecv Insertion	569
Session Agent Status Based on SIP Response	570
How It Works	570
Configuring Session Agent Status.	571
Session Agent Group Recursion	572
How It Works	573
Configuring Session Agent Recursion	573
Codec Stripping and Reordering	574
How It Works	574
In-Realm Codec Manipulation	577
Configuring Codec Stripping and Reordering	577
SIP REFER with Replaces	580
Configuring SIP REFER with Replaces	581
REFER Method Call Transfer	582
How it Works	583
Call Flows	584

Enabling SIP REFER Method Call Transfer.	586
REFER-Initiated Call Transfer	587
How it Works.	588
Basic Scenario	588
Notes	589
REFER Source Routing	592
Configuring Realm-Based REFER Method Call Transfer.	592
Configuring Session Agent- Based REFER Method Call Transfer	593
Specifying Policy Lookup	594
Registration Local Expiration	594
How It Works	594
Configuring Registration Local Expiration	595
Monthly Minutes-Based Call Admission Control	597
Caveats.	597
Configuring Monthly Minutes-Based CAC	597
HMR CPU Utilization Rates	598
Configuration Order	598
Expression Building Using Parentheses	598
Use of Escape Characters	598
New Comparison Types	598
Performing HMR on a Specific Header.	599
Multiple SIP Header Manipulation Rulesets	599
Configuring Multiple SIP HMR Sets	600
Registration Burst Rate Control	601
Configuring Registration Burst Rate Control	601
Message-Body Separator Normalization	602
Configuring Message-Body Separator Normalization.	602
ENUM: Large UDP Datagram Support for DNS NAPTR Responses	603
How It Works	603
Configuring Large UDP Datagram Support	603
SDP Address Correlation	604
Configuring SDP Address Correlation	604
SDP Insertion for (Re)INVITEs	608
How It Works	608
SIP: PRACK Interworking	610
How It Works	611
Configuring PRACK Interworking for a SIP Interface	613
NSEP Enhancements	615

NSEP for Session Agents	.615
Call Treatment.	.615
Log Level Changes	.615
Configuring NSEP Limits for Session Agents	.616
SIP Per User Subscribe Dialog Limit	.617
Limit Enforcement	.617
About Wildcarding	.617
Configuring an Enforcement Profile	.617
STUN Server	.619
About STUN Messaging	.619
STUN Server Functions on the Net-Net SBC	.621
RFC 3489 Procedures	.621
Configuring a STUN Server.	.622
SIP Globally Routable User Agent URIs	.623
Contact Header URI Replacement	.623
Record-Route Addition	.624
GRUU URI Parameter Name.	.624
Configuring SIP GRUU	.625
SIP Diversion to SIP-ISUP Interworking.	.626
Creating a SIP Profile	.626
Creating a SIP ISUP Profile	.627
Applying SIP Profiles and SIP ISUP Profiles	.628
HMR for SIP-ISUP	.631
Changes and Additions to Equality Operators	.631
Reserved Words	.632
Setting the Action Parameter.	.633
About MIME Rules	.633
Creating MIME Rules	.634
Creating MIME Headers	.637
About MIME ISUP Manipulation	.639
MIME ISUP Rules.	.643
SIP-ISUP Format Version Interworking	.650
Configuring SIP-ISUP Format Version Interworking	.651
Applying SIP ISUP Profiles	.652
SIP Proxy Subscriptions	.654
Topology Hiding.	.654
How It Works	.655
Configuring Proxy Sub events.	.656
SIP Duplicate SDP Suppression	.656
Configuring SIP Duplicate SDP Supression.	.657

SIP LMSD Interworking	657
About Alert-Info Processing.	658
About LMSD Processing	658
Configuring LMSD SIP Call Progress Tone Interworking.	660
Comma-Separated SIP Header Values	661
Configuring HMR Split Headers	662
Configuring HMR Join Headers.	663
Expedited Call Leg Release for Pre-empted Hairpin Calls	664
Accounting Considerations	664
Configuring Call Leg Release for Hairpin Calls	664
Expedited Media Termination of Calls with Abnormal Signaling Termination	665
Accounting Considerations	665
Configuring Graceful Termination	665
Wildcarded PUID in P-Associated-URI	666
Wildcarded Public User Identity	666
Configuring Wildcard PUID Registration	667
Alarm Synchronization (or Trap Event Monitoring)	667
Configuring Trap Event Monitoring	668
Configuring CAC, Policing, and Marking for non-Audio, non-Video Media	668
Custom RADIUS CDR VSAs for SIP	670
About User-Defined VSAs for SIP Calls	670
Configuring Custom CDR VSAs	672
Configuring Manipulation Strings	677
Trunk-Group VSA Generation	678
SIP HMR MIME Support	680
Availability	680
How It Works: Find and Replace All	680
Configuring a SIP Header Rule	684
Configuration Examples	685
Local Policy Session Agent Matching	691
How It Works	691
Configuring Local Policy Session Agent Mapping.	695
1532 - SIP Method Matching and To Header Use for Local Policies	696
Configuring TO Header Use for Local Routing	698
SIP Globally Routable User Agent URIs	699
Contact Header URI Replacement.	699
Record-Route Addition	699
GRUU URI Parameter Name	700
Configuring SIP GRUU	700

Whitelists	701
What is a Whitelist?	701
Configuring Whitelists	702
Configuration Exception	706
How Whitelists Work	706
Whitelist SNMP Traps	707
Whitelist Learning	708
Start/Stop Whitelist Learning	709
Displaying Rejected Messages	710
Conditional Bandwidth CAC for Media Release	711
About Conditional Bandwidth CAC for Media Release	711
Details and Conditions	712
Configuring a SIP Profile	713
Applying a SIP Profile	715
Configuring Require Header Option Tag	717
Full Via Header Replacement	718
Configure SIP Via Header Transparency	718
SIP-ISUP Format Version Interworking	719
Configuring SIP-ISUP Format Version Interworking	719
Applying SIP ISUP Profiles	721
IMS Charging Headers	723
Configuring P-Charging-Vector Processing for SIP Interfaces	727
IMS: P-CSCF Endpoint Identification Using Address and Port	729
Configuring P-CSFCF endpoint identification using address and port	729
1796 - SIP: Interim RADIUS Records for Recursive Attempts	730
Interim RADIUS Records for Recursive Attempts	730
Expedited Media Termination of Calls with Abnormal Signaling Termination	731
Accounting Considerations	731
Graceful Termination of Calls with Abnormal Signaling	731
SIP Diversion to SIP-ISUP Interworking	733
SIP Diversion to SIP-ISUP Interworking	733
SIP Registration Via Proxy	737
Considerations for Reg-Via-Key and Port Mapping	737
Request Routing	737
Configuring SIP Registration Via Proxy	738
SIP LMSD Interworking	740
About Alert-Info Processing	740
About LMSD Processing	740

Configuring LMSD SIP Call Progress Tone Interworking.	742
HMR for SDP Offer-Answer.	744
Configuring HMR SDP Offer-Answer	744
Wildcarded PUID in P-Associated-URI.	745
Wildcarded Public User Identity	746
Wildcarded PUID Registration.	746
Intrusion Detection System	747
License Requirements.	747
Rejected SIP Calls	747
Enabling Message Transfer to Specific Syslog Servers.	749
TCA Reporting of Denied Entries	750
CPU Load Limiting.	751
Denied Endpoints	751
SIP Proxy Subscriptions.	751
Topology Hiding	752
How It Works	752
Configuring Proxy Sub events	753
SIP SDP Insertion for Re-INVITES	754
How It Works	754
SIP REFER with Replaces	758
Configuring SIP REFER with Replaces	759
Replaces Header Support	762
Configuration Instructions	763
Replaces Header Support	764
Configuration Instructions	764
SIP Session Timer Feature	765
How the Session Timer Feature Works.	765
Enabling the Session Timer Feature	769
Session-Timeout RADIUS Value.	771
SIP Header Pre-Processing HMR	771
Enabling SIP Header Pre-Processing	772
LP Routing with ENUM provided SA Route.	774
Upgrades	774
Configuring Local Policy Routing with ENUM Provided Session Agents.	775
Simultaneous TCP Connection and Registration Cache Deletion	776
Configuring Simultaneous TCP Connection and Registration Cache Deletion	776
STUN Keep-Alive.	776
Platform Support.	777

License Requirements	777
How It Works	777
P-Asserted-ID Output to CDRs.....	782
P-Asserted-ID Output to CDRs.....	783
Configuring P-Asserted IDs in CDRs	783
P-CSCF: SIP Registration Event Package	784
How It Works	784
Consideration for HA Nodes.....	786
Enabling P-CSCF RegEventPackage.....	787
2423 - SIP-KPML to RFC 2833 Conversion for DTMF Events	787
SIP KPML to RFC 2833 Negotiation.....	787
RFC 2833 to SIP KPML Negotiation.....	787
Configuring KPML—2833 Interworking	788
SIP Subscriptions Management	788
Keep-Alive with CR/LF.....	791
Availability.....	791
Platform Support	791
License Requirements	791
How It Works	792
Session Timers Support	792
Configuring Session Timer Profiles.....	792
Associating Session Timer Profiles	794
Enabling CR/LF on SIP Interfaces.....	796
Session Load Balancer.....	797
Functional Overview	797
Balancing and Rebalancing	798
SLB Configuration	799
SBC Configuration	812
6 H.323 Signaling Services.....	815
Introduction.....	815
Signaling Modes of Operation.....	816
Before You Configure H.323.....	819
Peering Environment for H.323.....	819
Realm Bridging with Static and Dynamic Routing	820
Configuring Global H.323 Parameters.....	820
Configuring Global H.323 Parameters	820
Configuring Global Options	821

Configuring H.323 Interfaces	824
Configuration Timestamp	824
Configuring H.323 Interfaces	824
Configuring H.323 Service Modes	828
Configuring the H.323 Service Mode	828
Fast Start/Slow Start Translations	832
Fast Start to Slow Start Translation	832
Slow Start to Fast Start Translation	832
Configuration Prerequisites	833
Creating Media Profiles	833
Configuring Translations	837
RFC 2833: DTMF Interworking	839
About RFC 2833	839
About H.245 UII	839
About 2833 to H.245 UII Interworking	839
About DTMF Transfer	840
Preferred and Transparent 2833	840
Transparent 2833 Support	842
Basic RFC 2833 Negotiation Support	842
H.323 to H.323 Negotiation	843
Signal and Alpha Type Support	843
H.323 Endpoints	844
Translating H.245 UII to 2833 for H.323 Calls	844
Configuring DTMF Support	845
H.323 Registration Proxy	849
H.235 Authentication Transparency	849
Unique CSA Per Registered Gateway	849
Virtual Call Signaling Address	849
Virtual RAS Address	850
RAS Message Proxy	850
About Setting Port Ranges	850
Configuring H.323 Registration Proxy	851
H.323 Registration Caching	852
Caveats for Registration Caching	853
Configuration Requirements	853
Configuring the Gateway Interface for Registration Caching	854
H.245 Stage	855
Dynamic H.245 Stage Support	856
Configuring H.245 Stage Support	858
H.323 HNT for Outbound Sessions	859

Caveats.	.860
Configuring H.323 HNT for Outbound Sessions.	.861
H.323 Party Number-E.164 Support	.863
Signaling Only Operation.	.865
Configuring H.323 Signaling Only Operation	.867
Maintenance Proxy Function	.868
Configuring the Maintenance Proxy Function	.868
TCP Keepalive.	.870
Automatic Gatekeeper Discovery	.872
H.323 Alternate Routing	.873
Without Alternate Routing Enabled	.873
With Alternate Routing Enabled	.873
Configuring H.323 Alternate Routing.	.875
H.323 Bearer-Independent Setup	.876
H.323 BIS Disabled	.876
H.323 BIS Enabled	.876
Configuring H.323 BIS	.876
TOS Marking for H.323 Signaling.	.880
H.323 Options	.881
H.323 Stack Monitoring.	.883
Configuring H.323 Stack Monitoring	.884
H.323 Automatic Features.	.885
Alias Mapping	.885
Call Hold and Transfer	.885
Media Release for SS-FS Calls.	.891
H.323 and IWF Call Forwarding	.893
H.323 Accounting	.895
Configuring H.323 Accounting	.895
Additional Information	.904
About Payload Types	.904
H.323 Codec Fallback.	.908
Previous Behavior	.908
New Behavior	.909
Net-Net Central Instructions and Examples	.911
H.323/TCS Media Sample Size Preservation	.911
Configuring Media Sample Size Preservation	.912
H.323-TCS: H.245 Support for H.264 and G722.1	.915
Configuring	.915

H.323 LRQ Alternate Routing	919
How It Works	919
Caveats	920
H.323 Source Call Signaling Port	921
Net-Net Central Instructions and Examples	921
H.323 Switchover Behavior	921
STOP CDRs	921
Graceful Media Stream Termination	922
H.323 Switchover Configuration	922
ENUM/Local Route Table Support for H.323 and IWF	924
IWF Considerations	924
Configuring LRT Support	925
H.323 Switchover Behavior	926
STOP CDRs	926
Graceful Media Stream Termination	926
H.323 Switchover Configuration	926
H.323 - G726 Transcoding	928
H.323 - G726 Transcoding	928
International Peering with IWF and H.323 Calls	928
Configure International Peering for H.323 Calls	929
ENUM for H.323 and SIP-H.323 IWF	930
H.323 Stack Monitoring	930
Configure H.343 Stack Monitoring	930
Cause Code Completeness	931
H.323 CDR Generating Behaviors	931
Enable H.323 CDR Handling	932
H.323 Generating Behavior for OOS Session Agents	933
Enable Custom H.323 Cause Code for OOS Session Agents	933
SIP CDR Generating Behaviors	935
Enable SIP CDR Handling	935
H.323 Proprietary Identification Mechanism of a Registering Terminal	935
Requirements	935
How It Works	937
ACLI Instructions and Examples	939
Monitoring	940
H.323 Setup without Called Number	940
Caveats	941
ACLI Instructions and Examples	941

H.245 Separate Stack	942
Caveats.....	943
ACLI Instructions and Examples.....	943
H.264 FEC.....	945
Caveats.....	945
ACLI Instructions and Examples.....	946
7 IWF Services	947
Introduction.....	947
Access Network Application	947
Networking Peering Application.....	948
How It Works	948
SIP/H.323 Negotiation: H.323 Fast Start	948
SIP/H.323 Negotiation: H.323 Slow Start.....	950
Status and Codec Mapping	952
IWF Service Enhancements	956
SIP Redirect—H.323 LRQ Management	956
SIP INFO and DTMF UII Management	958
Mid-Session Media Change	958
Early Media	959
Display Name Mapping.....	959
IWF Ringback Support.....	959
H.323 Endpoint-Originated Call Hold and Transfer.....	966
Music On Hold	969
Conference	972
IWF Call Forwarding	973
Media Release for H.323 SS-FS Calls for IWF	974
Before You Configure	977
H.323 Configuration.....	977
SIP Configuration	977
The Role of Route (Local) Policy	978
Configuration Timestamp	979
Configuring Interworking	979
Configuring Interworking	979
DTMF Support	981
Configuring DTMF Support.....	982
RFC 2833: DTMF Interworking	986
About RFC 2833	986
About H.245 UII	986

About 2833 to H.245 UII Interworking	987
About DTMF Transfer.	987
Preferred and Transparent 2833.	988
Basic RFC 2833 Negotiation Support.	990
H.323 to SIP Calls	992
H.323 Non-2833 interworking with SIP	992
How H.323 to SIP Calls Work	993
Configuring RFC 2833 for DTMF Interworking	993
Configuring Transcoding Policy.	998
Payload Type Handling.	999
How It Works	1000
Configuring IWF Payload Type Handling.	1000
H.245 to 2833 DTMF Interworking Enhancements	1002
Configuring H.245 UII/SIP INFO-2833 DTMF Interworking	1003
Additional H.245 to 2833 DTMF Interworking	1007
Configuring RFC 2833 End Packets Only for Non-Signaled Digits Events	1007
DTMF Transparency for IWF	1008
Configuring DTMF Transparency for IWF	1008
Configuring the P-dual-info Header Appearance	1009
SIP Tel URI Support.	1010
Configuring SIP Tel URI Support	1012
IWF Inband Tone Option	1012
Configuring IWF Inband Tone Option	1013
RFC 3326 Support	1014
Default Mappings	1015
Configuring RFC 3326 Support	1017
IWF Privacy: Caller Privacy on Unsecure Networks	1020
About the Presentation Indicator.	1020
H.323 to SIP IWF Call.	1020
SIP to H.323 IWF Call.	1022
IWF Privacy: Caller Privacy on Secure Connections	1024
H.323 to SIP IWF.	1024
H.323 to SIP.	1024
SIP to H.323.	1025
Trunk Group URIs	1028
Terminology.	1028
Trunk Group URI Parameters	1028
How It Works	1032
Before You Configure	1033

Setting the Trunk Group URI Mode for Routing	1034
Configuring a Session Agent for Trunk Group URIs	1035
Configuring a Session Agent Group for Trunk Groups URIs	1037
Setting a Trunk Group Context in a Realm	1038
Trunk Group URIs with SIP Interface Registration Caching	1038
Options for Calls that Require the IWF	1040
Configuring Options for Calls that Require the IWF	1041
IWF Privacy for Business Trunking	1045
A Call Originating in H.323	1046
A Call Originating in SIP	1047
Net-Net Central Instructions and Examples	1049
IWF COLP/COLR Support	1050
How It Works	1050
Net-Net Central Instructions and Examples	1052
IWF Privacy Extensions for Asserted Identity in Untrusted Networks	1053
IWF Call Originating in H.323	1053
Before You Configure	1056
Net-Net Central Instructions and Examples	1056
IWF RAS Registration Failure Code Mapping	1057
Supported Status Codes	1057
Net-Net Central Instructions and Examples	1058
IWF Codec Negotiation: H.323 Slow Start to SIP	1058
Configuring Codec Negotiation	1059
IWF: H.245 Signaling Support for G.726	1059
How It Works: IWF	1060
Configuring Signaling Support for G.726	1060
Customized G.729 Support	1069
How It Works	1069
Configuring Customized G.729 Support	1070
IWF Codec Renegotiation for Audio Sessions	1073
Codec Request Change from the SIP Side	1073
Codec Request Change from the H.323 Side	1073
Exceptional Cases	1073
Enabling IWF Codec Renegotiation	1074
SIP Early Media Suppression	1074
How It Works	1075
About the Early Media Suppression Rule	1077
Selective Early Media Suppression	1077
Configuring Early Media Suppression	1078

IWF Codec Renegotiation for Audio Sessions	1080
Codec Request Change from the SIP Side	1080
Codec Request Change from the H.323 Side	1080
Exceptional Cases	1080
Enabling IWF Codec Renegotiation.....	1081
Reliable Response Interworking	1082
LMSD SIP to IMS SIP Interworking Enhancement	1083
 8 MGCP/NCS Signaling Services	 1085
Introduction	1085
MGCP/NCS Overview	1085
MGCP/NCS Network Topology	1087
MGCP/NCS Configuration Overview.....	1087
Before You Configure	1088
Configuring MGCP/NCS Signaling Services	1088
DNS Authentication	1091
DNS Authentication Configuration Prerequisites	1092
Configuring DNS Authentication	1092
Endpoint Profile	1093
Endpoint Number Computation	1094
Endpoint Translation Mode	1095
Configuring MGCP Endpoint Profile	1097
MGCP/NCS Options	1100
MGCP Options	1100
Send Media Only.....	1102
Signaling the Source IPv4 Address of Endpoints for 911 Services	1103
Loose Authentication	1103
Configuring MGCP Options	1103
MGCP/NCS 1:1 IPv4 Address Mapping	1105
Configuring MGCP/NCS 1:1 IPv4 Address Mapping	1106
MGCP/NCS SuperNAT	1107
Configuring MGCP/SuperNAT	1107
Call Agent Redundancy	1109
Call Agent Redundancy Configuration Overview	1109
Configuring Call Agent Redundancy.....	1110
MGCP Call Agent Switchover	1111
Configuring Call Agent Switchover.....	1112
Enhanced Roaming (IP Address Carrying)	1113

Configuring Enhanced Roaming	1113
MGCP Sans Media	1113
Configuring MGCP Sans Media	1114
MGCP/NCS X-Keepalives	1115
Configuring X-Keepalives	1115
MGCP Congestion Control	1117
Configuring MGCP Congestion Control	1117
Graceful Stateful Backoff	1118
MGCP Endpoint Aging	1120
How It Works	1120
Configuring Endpoint Aging	1121
Endpoint Aging Optimization	1122
Endpoint Removal	1125
MGCP Restricted Latching	1127
Configuring Restricted Latching	1127
MGCP Port Mapping	1128
How It Works	1128
Configuring MGCP Port Mapping	1129
Reordering and Stripping Multiple Codecs for MGCP	1131
Reordering Codecs	1133
Deleting Codecs	1135
MGCP Configurable CPU Sample Rate	1135
How It Works	1135
Configuring MGCP Sampling Rate	1136
MGCP: NAT Traversal without Audit Interval	1137
Configuring MGCP to Allow NAT Traversal Without Audits	1138
 9 DNS Application Layer Gateway Service	 1141
Introduction	1141
Overview	1141
Configuring DNS ALG Service	1142
Before You Configure	1142
Configuring DNS ALG	1143
DNS Transaction Timeout	1146
Enabling DNS Transaction Timeout	1146
H.248 ALG	1147
Sample Application	1147

Gateway Masquerading	1147
Handoff Support	1148
Licensing	1149
Configuring H.248.	1149
DNS Transaction Timeout	1155
Enabling DNS Transaction Timeout	1155
Recursive ENUM Queries.....	1155
Configuring Recursive Query.....	1156
Unique DNS Server per VLAN with OLIP	1156
DNS ALG Message Throttling.....	1156
Configuration Instructions	1157
 10 Session Routing and Load Balancing	 1161
Introduction	1161
Routing Overview	1161
Session Agents, Session Groups, and Local Policy	1161
About Session Agents	1162
SIP Session Agents	1162
H.323 Session Agents	1163
Managing Session Agent Traffic	1163
About Session Agent Groups.....	1164
About Local Policy	1165
SIP Session Routing	1167
About the Ingress Realm	1168
About the Egress Realm	1168
About SIP Redirect	1168
SIP Method Matching and To Header Use for Local Policies	1169
SIP Methods for Local Policies.....	1169
Configuring SIP Methods for Local Policy Matching.....	1169
Configuring TO Header Use	1170
H.323 Session Routing	1172
Egress Stack Selection.....	1172
Registration Caching.....	1173
Gatekeeper Provided Routes	1173
Load Balancing	1176
Configuring Routing	1177
Configuration Prerequisite	1177
Configuration Order	1177

Configuration Timestamp	1177
Creating a Media Profile	1178
Creating a Session Agent	1181
Configuring a SIP Session Agent	1182
Configuring an H.323 Session Agent	1195
Configuring Session Agent Groups	1203
Configuring Routes	1206
Configuring Holidays	1211
Route (Local) Policy Matching for Parent Realms	1212
Configuring Route (Local) Policy Matching	1212
Answer to Seizure Ratio-Based Routing	1213
How It Works	1213
Configuring ASR Constraints	1214
SIP Session Agent Continuous Ping	1215
How It Works	1215
Configuring the Ping Mode for a Session Agent	1216
Session Agent Egress Realm	1216
How It Works	1217
Configuring a Session Agent Egress Realm	1217
ENUM Lookup	1218
How ENUM Works	1218
About the Net-Net SBC ENUM Functionality	1219
ENUM Failover and Query Distribution	1219
Failover to New ENUM Configuration	1219
ENUM Server Operation States	1220
Server Availability Monitoring	1220
ENUM Server IP Address and Port	1220
Configuring ENUM Failover and Query Distribution	1220
Source URI Information in ENUM Requests	1222
Configuring OPT-RR Additional Records Sent to ENUM Servers	1222
Caching ENUM Responses	1223
Adding Additional Records to the Local Cache	1224
Operation Modes	1224
Configuring ENUM	1226
Custom ENUM Service Type Support	1228
Configuring Custom ENUM Support	1228
Configuring Local Policy Attributes	1229
CNAM Subtype Support for ENUM Queries	1231
CNAM Unavailable Response	1232

SIP Profile Inheritance	1232
Configuring CNAM Subtype Support	1232
ENUM/Local Route Table Support for H.323 and IWF	1233
IWF Considerations	1233
Configuring LRT Support	1234
P-CSCF Redundancy with Implicit Service Route	1235
Tracking UE per P-CSCF	1235
Registration Invalidation	1235
Configuring P-CSCF Redundancy with Implicit Service Route	1236
Route Recursion Override	1236
How It Works	1237
Configuring Route Recursion Override	1237
Codec Reordering	1238
How It Works	1238
Configuring Codec Reordering	1239
QoS Based Routing	1240
Configuring QoS Constraints	1241
Multi-Stage Local Policy Routing for SIP	1243
Routing Stages	1243
Network Applications	1243
Multistage Routing Conceptual Example	1244
Multistage Routing Example 2	1245
Customizing Lookup Keys	1246
Multistage Routing Lookup Termination	1247
SIP Session Agent DNS-SRV Load Balancing	1248
Configuring Session Agent DNS-SRV Load Balancing	1249
Route (Local) Policy Matching for Parent Realms	1250
Configuring Route (Local) Policy Matching	1250
Source URI Information in ENUM Requests	1251
Configuring OPT-RR Additional Records Sent to ENUM Servers	1252
ENUM Failover and Query Distribution	1252
Failover to New ENUM Configuration	1252
ENUM Server Operation States	1253
Server Availability Monitoring	1253
ENUM Server IP Address and Port	1253
Configuring ENUM Failover and Query Distribution	1254
Source URI Information in ENUM Requests	1255
Configuring OPT-RR Additional Records Sent to ENUM Servers	1255

Local Route Tables (Onboard ENUM)	1256
How It Works	1256
Net-Net Central Instructions and Examples	1257
Multi-Stage Local Policy Routing for SIP	1259
Routing Stages	1259
Network Applications	1259
Multistage Routing Conceptual Example	1259
Multistage Routing Process Example	1260
Customizing Lookup Keys	1262
Multistage Routing Lookup Termination	1262
Unique Header Manipulation Rule Regex Patterns	1264
Manipulation Pattern Per Remote Entity	1265
Reject Action	1265
Log Action	1266
Changes to Storing Pattern Rule Values	1267
Removal of Restrictions	1267
Name Restrictions for Manipulation Rules	1267
New Value Restrictions	1268
LRT Entry Matching	1268
2349- LRT String Lookup	1269
Directed Egress Realm from LRT/ENUM - 2473	1269
SIP Embedded Route Header	1270
LRT Lookup Key Creation	1272
2274 - Retargeting LRT/ENUM-based Requests	1272
11 Address Translation	1275
Introduction	1275
Address Translation Configuration Overview	1276
Configuring Address Translation Profiles and Rules	1278
Configuring Translation Profiles and Rules	1278
Other Translations	1285
12 Admission Control and Quality of Service Reporting	1287
Overview	1287
About Call Admission Control	1287
Bandwidth-Based Admission Control	1287
Session Capacity- and Rate-based Admission Control	1290

Asynchronous SIP-Diameter Communication	1290
..... Serialized Diameter Messaging	1292
Configuring the External Bandwidth Manager	1293
Bandwidth CAC Fallback Based on ICMP Failure	1293
Enabling Bandwidth CAC Fallback	1294
Bandwidth CAC for Aggregate Emergency Sessions	1297
Configuring Bandwidth CAC for Aggregate Emergency Sessions	1297
Configuring Admission Control	1298
Configuration Timestamp	1298
Adding a Session Agent	1298
Configuring Realm Bandwidth	1304
Conditional Bandwidth CAC for Media Release	1307
About Conditional Bandwidth CAC for Media Release	1307
Details and Conditions	1307
Configuring a SIP Profile	1309
Applying a SIP Profile	1312
Configuring Require Header Option Tag	1314
Configuring CAC, Policing, and Marking for non-Audio, non-Video Media	1314
About QoS Reporting	1317
Overview	1317
Aggregate Session Constraints: Nested Realms	1323
Impact to Other Session Constraints and Emergency Calls	1323
Configuring Session Constraints for Nested Realms	1324
Configuring QoS	1325
Configuring QoS	1325
Configuring Network Management Controls	1326
How It Works	1326
Configuring Network Management Controls	1329
Network Management Control Dialed Number Enhancement	1333
Configuring the Account Configuration and Account Servers	1334
Configuring the Account Configuration and Account Servers	1334
Aggregate Session Constraints Per Realm	1337
Configuring Session Constraints per Realm	1337
Session Agent Minimum Reserved Bandwidth	1338
How It Works	1338
Configuring Minimum Reserved Bandwidth For Session Agents	1339
Aggregate Session Constraints for SIP	1339
Configuring Session Constraints	1340

Shared CAC for SIP Forked Calls	1344
Bandwidth Sharing Scenarios	1344
Configuring Bandwidth Sharing	1345
TLS Call Recording	1348
Configuring TLS Call Recording	1348
13 Static Flows	1351
Introduction.....	1351
About Static Flows	1351
About Network Address Translation ALG.....	1352
Configuring Static Flows.....	1353
Basic Static Flow Configuration Overview	1353
Configuration Timestamp	1354
Configuring Static Flows	1354
Editing Static Flows	1355
14 High Availability Nodes	1359
Overview	1359
Net-Net Central HA Cluster	1359
Terminology	1359
HA Process	1360
Single Master with Multiple Replica Strategy	1362
Multi-Member Clustering	1362
Two-Member Clustering	1365
Data Synchronization Scenarios	1368
Adding Member to a Cluster	1368
Member of a Cluster Fails or is Shutdown	1368
Member Rejoining a Cluster	1369
High Availability Scenarios	1370
Setup a Cluster	1370
Shutdown a Cluster	1371
Retiring Cluster Member	1371
Network Partition in Two Node Cluster.....	1372
Network Partition in a Three Node Cluster	1372
15 Security	1373
Introduction.....	1373
Security Overview.....	1373

Denial of Service Protection	1374
Levels of DoS Protection	1375
About the Process	1376
Trusted Path	1377
Untrusted Path	1377
Static and Dynamic ACL Entry Limits	1378
Dynamic Deny for HNT	1378
Host and Media Path Protection Process	1379
Session Border Controller Access Control	1379
Media Access Control	1380
Host Path Traffic Management	1380
Traffic Promotion	1380
Malicious Source Blocking	1380
Blocking Actions	1381
Protecting Against Session Agent Overloads	1381
DoS Flow Classification Allocation	1381
Configuration Overview	1382
Changing the Default Net-Net SBC Behavior	1382
Configuration Timestamp	1383
Dynamic Demotion for NAT Devices	1384
Enabling Dynamic Demotion of NAT Devices	1384
Configuring DoS Security	1385
Host Access Policing	1385
Basic Access Control for a Realm	1386
Managing RSIPs	1390
Configuring Managing RSIPs for DoS	1390
Configuring Overload Protection for Session Agents	1390
Configuring DoS Flow Classification Allocation	1391
Media Policing	1393
Policing Methods	1394
Configuration Notes	1394
Configuring Media Policing for RTP Flows	1395
Configuring Media Policing for Static Flows	1396
RTP Payload Type Mapping	1396
ITU-T to IANA Codec Mapping	1397
TCP Synchronize Attack Prevention	1397
About SYN	1397
Configuring TCP SYN Attack Prevention	1398
Transport Layer Security	1399
The Net-Net SBC and TLS	1399

TLS Features	1400
Domestic and International Versions	1400
Compliance	1401
Supported Encryption	1401
Signaling Support	1402
DoS Protection	1402
Endpoint Authentication	1403
Viewing TLS Support Information	1403
ACLI Process Overview	1403
Viewing TLS Certificate Records	1404
Configuring a TLS Profile	1409
Viewing TLS Profiles using Net-Net Central	1410
Applying a TLS Profile	1412
Reusing a TLS Connection	1412
Keeping Pinholes Open at the Endpoint	1412
Viewing Certificates	1413
Denial of Service for TLS	1414
Configuring DoS for TLS	1415
TLS Key Usage and Other Enhancements	1418
Key Usage Control	1418
TLS Version	1419
TLS Session Caching	1420
Enabling TLS Session Caching	1421
OCSP Support for SIP-TLS	1421
Viewing Certificate Status Profile Data	1421
Viewing TLS Profile Data	1422
Password Strength	1423
Password Policy	1423
System Access Control Lists	1425
Configuring System Access Control Lists	1426
IPSec Manual Keying	1427
Supported Protocols	1427
IPsec Implementation	1428
Outbound Packet Processing	1429
Inbound Packet Processing	1431
HA Considerations	1432
Packet Size Considerations	1432
IPsec Application Example	1433
Configuring Security Associations	1433
Configuring Security Policies	1439

National Security and Emergency Preparedness for SIP	1442
How It Works	1442
Configuring ETS	1445
IDS Reporting	1451
IDS Licensing	1451
Basic Endpoint Demotion Behavior	1452
Endpoint Demotion Reporting	1452
Endpoint Demotion SNMP Traps	1453
Endpoint Demotion Syslog Message	1453
Configuring Endpoint Demotion	1454
Endpoint Demotion due to CAC Overage	1454
Configuring Endpoint Demotion on CAC Failures	1455
SRTP-SDES Configuration	1456
Configuring an SDES Profile	1457
Configuring Media Security Policy	1459
Configuring Inbound Parameters	1461
Configuring Outbound Parameters	1462
FIPS 140-2 Level 2	1463
Configuring FIPS 140-2 Level2	1463
SSH Public Key Support	1464
TLS IPv6	1473
Configuring TLS for IPv6	1473
Unreachable OCSR	1474
Online Certificate Status Protocol	1474
Caveats	1475
Creating a Certificate Status Profile using the ACLI	1475
High Capacity SRTP SDES	1477
SRTP Support	1478
Configuring SRTP Support	1478
Intrusion Detection System	1479
License Requirements	1479
Rejected SIP Calls	1479
Enabling Message Transfer to Specific Syslog Servers	1481
TCA Reporting of Denied Entries	1482
CPU Load Limiting	1483
Denied Endpoints	1483
Anonymous Message Processing	1483
Processing Anonymous From- Headers	1484
Enabling Regulatory-Compliant Processing of Anonymous From-Headers	1484

250000 Trusted Endpoints	1485
SBC Host Certificates SNMP	1486
Configuration Instructions	1486
16 Lawful Intercept	1487
Introduction	1487
Recommendations	1488
Interoperability with SS8	1489
Interoperability with Verint	1489
Interoperability Using a Dynamic Trigger by CMS	1490
Interoperability Using ALIP	1490
SS8 Xcipio Global Configuration	1491
Before You Configure	1491
Configuring SS8 Xcipio Attributes	1491
INI1 Interface Configuration	1493
INI2 Interface Configuration	1495
INI3 Interface Configuration	1497
LI Pre-Provisioned Targets to 30K	1499
P-DCS-LAES: SIP Session Interception Mode	1499
Configuring Hairpin Reporting Algorithm	1501
Configuring Multiple Header Mode Algorithm	1504
17 External Policy Servers	1507
Introduction	1507
Call Admission Control	1507
Implementation Features	1508
Bandwidth Negotiation	1509
COPS Connection	1509
COPS Failures	1509
Net-Net High Availability Support for COPS	1510
COPS Debugging	1511
Configuring COPS	1512
Creating a Policy Server	1512
Configuring Realms	1513
Configuring Profiles	1514
Diameter: Subscriber Information Enhancement	1515
About the Subscription-Id AVP	1515
Configuring Subscription-Id AVP Support	1516

Connectivity Location Function	1517
CLF Behavior	1517
CLF Failures	1519
CLF Emergency Call Handling	1519
HA Functionality	1520
CLF Debugging	1521
Configuring CLF	1521
Configuring CLF	1521
Diameter: CLF and RACF	1524
Diameter Connection	1524
Diameter Heartbeat	1525
Diameter Failures	1526
Application IDs	1527
Diameter Transport-Class/Optional AVP and Application Mode Support	1527
Optional AVP Support	1527
New Application Mode Setting	1528
Configuring Optional AVPs	1529
Configuring Service Class Options	1531
Setting the Application Mode	1533
Diameter: Bandwidth-Based Call Admission Control	1534
Implementation Features	1534
Bandwidth Negotiation	1535
Session Lifetime	1535
Net-Net High Availability Support for CAC	1535
Diameter AAR Query Post SDP Exchange	1536
Configuring AAR Post SDP Exchange	1536
Diameter: Configuring RACF	1538
Configuring the External Policy Server	1538
Configuring Realms for Diameter Support	1540
Configuring Media Profiles for CAC	1541
Diameter: Connectivity Location Function	1541
CLF Behavior	1542
CLF Failures	1543
CLF Emergency Call Handling	1543
HA Functionality	1544
Diameter: Configuring CLF	1545
Configuring External Policy Server for CLF	1545
Configuring the SIP Interface for CLF	1548
Diameter e2	1548

How It Works: CLF	1548
How It Works: RACF Experimental Result Handling	1549
About Realms and e2 Enhancements	1549
Configuring Diameter e2	1550
Diameter Gq Proxy Interface and Bit Enhancements	1552
Configuring Diameter Gq	1552
Diameter Rx Enhancements	1554
How it Works	1554
Configuring Rx Mode	1555
Diameter Transport-Class/Optional AVP and Application Mode Support	1557
Optional AVP Support	1557
New Application Mode Setting	1558
Configuring Optional AVPs	1559
Setting the Application Mode	1560
Diameter Media Policing Enhancements	1561
About Exact Bandwidth Definition	1562
Internal Media Policing On-Off Switch	1562
Setting Exact Bandwidth and Disabling Media Policing	1562
Diameter e2 Realm Identification	1564
How It Works	1564
Configuring e2 Realm Identification	1565
Diameter Permit on Reject	1566
TACACS+	1567
Introduction	1567
Authentication	1568
Authentication Message Exchange	1568
Authorization	1569
Accounting	1569
TACACS+ Configuration	1570
Managing TACAS+ Operations	1573
Diameter: Subscriber Information Enhancement	1574
About the Subscription-Id AVP	1574
Configuring Subscription-Id AVP Support	1575
DIAMETER: CLF e2 Interface User-Name AVP Support	1576
How It Works	1576
Configure DIAMETER Permit on Reject	1577
Diameter based RACF & CLF IPv6 Support	1578
IPv6 Addresses in UTF-8 Format	1578
Framed-IPv6-Prefix AVP	1578

Bandwidth Allocation Compensation for IPv6	1578
Adding a Standard Packet Rate	1579
Signaling Only P-CSCF	1579
Asynchronous SIP-Diameter Communication	1583
..... Serialized Diameter Messaging	1585
Configuring the External Bandwidth Manager	1586
FQDN as Address for External Policy Server	1587
FQDN Configuration	1587
Diameter Policy Server High Availability	1587
External Policy Server HA Cluster	1587
Standby Server Prioritization	1587
Server States	1587
HA Cluster Refresh	1588
DNS Failure	1588
Policy Server Failover	1588
Configure External Bandwidth Manager	1589
Multi-tiered Policy Server Support	1590
Messaging Details	1590
Multi-Tiered Policy Server Support	1591
FQDN as Address for External Policy Server	1591
FQDN Configuration	1591
Diameter Policy Server High Availability	1592
External Policy Server HA Cluster	1592
Standby Server Prioritization	1592
Server States	1592
HA Cluster Refresh	1592
DNS Failure	1593
Policy Server Failover	1593
Configuring the External Bandwidth Manager	1593
CLF/Diameter e2 Error Handling	1594
Result-Code AVP	1595
Experimental-Result-Code AVP	1595
Diameter CLF/e2 Error Bypass	1595
Configuration Instructions	1595
18 IMS Support	1597
Introduction	1597
IMS Access Border Functions	1598
P-CSCF Functions	1598

A-BGF Functions	1598
Resource and Admission Control (RACS) Functions	1599
IMS Interconnect Border Functions	1600
Interworking Function (IWF)	1600
Interconnect Border Control Function (I-BCF)	1600
Interconnect-Border Gateway Function (I-BGF)	1600
IMS Path and Service Route Header Support	1600
Path Header	1600
Service Route Header	1601
Summary	1601
Configuring IMS Path and Service Route Header Support	1602
IMS Support for Private Header Extensions for 3GPP	1603
P-Associated-URI Header	1603
P-Asserted-Identity Header	1603
P-Called-Party-ID Header	1605
IMS Charging Headers	1605
SIP Charging Function Address Enhancements for IMS	1608
Notes on Prior Behavior	1608
How It Works	1609
Configuring SIP Charging Function Address Enhancements	1610
RADIUS Accounting of Charging Headers	1611
P-Visited-Network-ID Header	1612
SIP IMS and Privacy	1613
Setting the PPI to PAI Conversion Mode	1613
Recurse 305-Only	1614
Redirect Action Process	1614
Redirect-Action Set to "Proxy"	1615
Redirect-Action Set to "Recurse"	1616
Redirect-Action Set to "Recurse-305-Only"	1617
Configuring "Recurse 305 Only"	1618
Implicit Service Route	1620
Enabling Implicit Service-Route Support	1620
Service Route Use	1621
Surrogate Registration	1623
Integrating with IMS	1623
How it Works	1624
Routing Calls from the IMS Core	1625
Routing Calls from the IP-PBX	1626
Configuring Surrogate Registration	1626

SIP Surrogate Registration Enhancements	1630
Previous Behavior	1630
New Behavior	1630
Configuring	1630
SNMP Support	1631
IMS Charging Vector Mode Adaptation	1632
Configuring P-Charging-Vector	1632
IMS: P-CSCF Endpoint Identification Using Address and Port	1633
Configuring P-CSCF Endpoint Identification	1633
IMS-AKA	1634
Requirements	1634
Enabling IMS-AKA	1634
E-CSCF Support	1639
Service URN Support	1639
E-CSCF Configuration Architecture	1639
Configuring an NMC for E-CSCF Use	1641
Configuring the Destination ID	1643
Configuring a Local Policy for E-CSCF Use	1643
Configuring Policy Attributes for E-CSCF Use	1644
SIP Surrogate Registration for Non-Register Requests	1644
Configuring SIP Surrogate Registration	1645
 19 Transcoding	 1647
Introduction	1647
Hardware Support: the Net-Net SBC TCU	1647
How It Works	1649
Transcoding Example	1650
Supported Codecs	1651
EVRC Family of Codecs	1651
EVRC-A Codec for Transcoding	1652
Default settings for EVRC encoding:	1653
EVRC Configuration	1653
EVRC-B Codec for Transcoding	1653
Default/fixed settings for EVRCB encoding:	1654
EVRCB Configuration	1654
EVRC Family of Codecs	1655
EVRC-A Codec for Transcoding	1655
Default settings for EVRC encoding:	1656

EVRC Configuration	1656
EVRC-B Codec for Transcoding	1656
Default/fixed settings for EVRCB encoding:	1658
EVRCB Configuration	1658
Configuring Transcoding	1658
Configuration Timestamp	1658
Adding New Transcoding Policies	1659
Allowing, Adding, and Removing Codecs	1660
Specifying an Order for Codecs	1663
Applying a Transcoding Policy to a Realm	1665
AMR-NB and EVRC0 Transcodable Codecs	1667
AMR-NB	1667
EVRC0	1667
Licensing	1669
Media Type Subnames	1669
SDP Parameter Matching	1669
Using Subnames with Transcoding Policies	1670
Configuring a Media Type Subname	1671
AMR-NB and EVRC0 Transcodable Codecs	1671
AMR-NB	1671
EVRC0	1672
Transcoding Capacity	1673
Licensing	1673
H.323 - G726 Transcoding	1673
Network Architecture	1675
 20 Diameter Director Configuration	 1676
Diameter Director Elements	1676
Diameter Director Interface	1676
Supported Applications	1676
Origin Host	1677
Diameter Director Ports	1677
Licensing	1677
Configuring Diameter Director Interfaces	1678
Diameter Director Agent	1680
Watchdog Messages	1681
Configuring Diameter Director Agents	1681
Capabilities Exchange Messaging	1683
Diameter Heartbeat (DWR/DWA)	1685

Disconnect Peer Messaging	1685
Message Type Verification	1686
Relay Mode	1686
Routing	1687
Diameter Message Reception	1687
Routing Criteria	1687
Routing Evaluation Data Types	1688
Routing Decision Process	1689
Grouped AVP Routing	1690
Dynamic Routing	1691
Policy Rejection	1691
Configuring Root Diameter Director Policies	1692
Diameter Director Group Recursive Routing	1696
Configuring Recursive Routing	1697
Diameter Director Groups	1698
Diameter Director Group Strategy	1698
Diameter Director Group States	1699
Diameter Director Group startup and shutdown	1699
Configuring Diameter Director Groups	1700
Manipulation Application	1701
Manipulation Rule	1701
Configuring Diameter Manipulation	1702
Message Manipulation Caveats	1705
CER/CEA	1705
AVP Manipulation Caveats	1707
.....Origin-State-Id AVP	1707
.....Experimental-Result Code Manipulation	1707
Anonymous Diameter Agent Blocking	1708
Supported Commands for KPI Tracking	1710
Supporting Configurations	1712
DNS for Diameter Director Agent Hostname Resolution	1712
Configuring the Network Interface	1713
CPU Load Limiting	1714
Configuring Diameter Director Interfaces for Load Limiting	1714
Diameter Director On-Demand Peering	1715
Configuring On-Demand Peering for Diameter Director	1715
Session Statefulness	1716
Configuring Stateful Policies	1716

Diameter Director Generic Manipulation	1717
Adding Manipulation Rules	1717
Configuring Diameter Manipulation Rules	1717
LIR/LIA Lookup to Home Subscriber Server from I-BCF	1718
Home Subscriber Server	1718
Local Policy	1718
Configuration Instructions	1719
 Acronym List	 1721
General Use Acronyms	1721
A	1721
B	1722
C	1722
D	1724
E	1725
F	1725
G	1725
H	1726
I	1726
J	1727
K	1727
L	1727
M	1728
N	1729
O	1729
P	1730
Q	1731
R	1731
S	1732
T	1733
U	1734
V	1734
W	1735
X	1735
Y	1735
Z	1735
 Signaling Protocol Acronyms	 1735
H.323	1735

About this Guide

Overview

The *Oracle Communications Session Element Manager Configuration Guide* provides information about the following:

- Basic concepts that apply to the key features and abilities of your Net-Net System
- Information about how to load the Net-Net system software image you want to use and establish basic operating parameters
- Configure system-level functionality for the Net-Net system
- Configure all components of the Net-Net SBC

Configuring Different Versions of Net-Net SBCs

You can use Net-Net Central to configure different versions of Net-Net SBCs. You might notice a difference in screen display and available functionality depending on the Net-Net SBC version. Some features and functionality documented in this guide might not be available.

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by	admin@console
Last modified date	2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2_UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Revision History

This section contains a revision history for this document.

Date	Revision Number	Description
December 11, 2012	Revision 1.00	Initial version

Introduction

This chapter introduces some basic concepts that apply to the key features and abilities of your Net-Net System. It is necessary that you understand the information included in this chapter to comprehend the ways to configure your Net-Net SBC. This chapter only provides a high level overview of some important Net-Net SBC concepts. Please refer to each chapter for complete descriptions of these concepts and the procedures for their configuration.

Realms

What Is a Realm?

A realm is a logical way of identifying a domain, a network, a collection of networks, or a set of addresses. Realms are used when a Net-Net SBC communicates with multiple network elements over a shared intermediate connection. Defining realms allows flows to pass through a connection point between two networks.

From an external perspective, a realm is a collection of systems that generates real-time interactive communication sessions comprised of signaling messages and media flows, or a group of multiple networks containing these systems. These systems may be session agents such as call agents, softswitches, SIP proxies, H.323 gatekeepers, IP PBXs, etc., that can be defined by IPv4 addresses. These systems can also be IP endpoints such as SIP phones, IADs, MTAs, media gateways, etc.

From an internal perspective, a realm is associated with Net-Net system configurations to define interfaces and resources in a logical way. Realms are used to support policies that control the collection of systems or networks that generate media sessions. Realms are referenced by other configuration elements in order to support this functionality across the protocol the Net-Net SBC supports and to make routing decisions.

Nested Realms

Nested Realms is a Net-Net SBC feature that supports hierarchical realm groups. One or more realms may be nested within higher order realms. Realms and sub-realms may be created for media and bandwidth management purposes. This feature supports:

- Separation of signaling & media on unique network interfaces
- Signaling channel aggregation for Hosted IP Services applications
- Configuration scalability
- Per-realm media scalability beyond single physical interface capacity
- Nested bandwidth admission control policies

Session Agents and Session Agent Groups

What Is a Session Agent?

A session agent defines an internal signaling endpoint. It is an internal next hop signaling entity that applies traffic shaping attributes to flows. For each session agent, concurrent session capacity and rate attributes can be defined. Service elements such as gateways, softswitches, and gatekeepers are defined automatically within the Net-Net SBC as session agents. The Net-Net SBC can also provide the defined session agents.

SIP session agents

SIP session agents can include the following:

- Softswitches
- SIP proxies
- Application servers
- SIP gateways

H.323 session agents

H.323 session agents can include the following:

- gatekeepers
- gateways
- MCUs

Why You Need Session Agents

You can use session agents to describe next or previous hops. You can also define and identify preferred carriers to use for traffic coming from session agents. This set of carriers is matched against the local policy for requests coming from the session agent. Constraints can also be set for specific hops.

In addition to functioning as a logical next hop for a signaling message, session agents can provide information regarding next hops or previous hops for SIP packets, including providing a list of equivalent next hops.

How to Use Session Agents

You can use session agents and session agent groups (along with local policies) to define session routing for SIP and H.323 traffic. You can associate a realm with a session agent to identify the realm for sessions coming from or going to the session agent.

What is a Session Agent Group?

A session agent group contains individual session agents bundled together, as well as other session agent groups. A SAG indicates that its members are logically equivalent and can be used interchangeably. This allows for the creation of constructs like hunt groups for application servers or gateways. Session agent groups also assist in load balancing among session agents.

Session agent groups can be logically equivalent to the following:

- Application server cluster
- Media gateway cluster
- Softswitch redundant pair

- SIP proxy redundant pair
- Gatekeeper redundant pair

High Availability (HA)

Net-Net SBCs are deployed in pairs to deliver continuous high availability (HA) for interactive communication services. The HA design guarantees that no calls are dropped in the event of any single point failure. Furthermore, the Net-Net SBC HA design provides for full media and call state to be shared across an HA node. The solution uses a VRRP-like design, where the two systems share a virtual MAC address and virtual IPv4 address for seamless switchovers.

In the HA pair, one Net-Net SBC is the primary system, and is used to process signaling and media traffic. The backup system remains fully synchronized with the primary system's session status. The primary system continuously monitors itself for connectivity and internal process health. If it detects service-disrupting conditions or degraded service levels, it will alert the backup Net-Net SBC to become the active system.

Introduction

Overview

Net-Net Central provides a centralized framework for working with Net-Net SBCs (devices). It lets you add the devices deployed in your network, configure those devices, view device information, perform administrative functions, and supports additional licensed applications.

Net-Net Central delivers:

- Scalability: User sessions are load balanced across a cluster of servers
- On-demand configuration data: Load configurations on-demand to provision locally and then push your changes to the device
- High availability: With the clustering of NNC servers any single point of failure is minimized
- Rich thin client: Client interface is delivered through your browser

In this Chapter

This chapter contains information you should review before you start configuring Net-Net SBCs using Net-Net Central. It includes such information as how to:

- install Net-Net Central
- load configuration
- save and activate to the device
- set up NTP synchronization
- configure RADIUS authentication

Prior to configuring your Net-Net SBC for service, Acme Packet recommends that you review the information and procedures in this chapter. It contains information that requires you to use the Acme Packet command line interface (ACLI) for some of the procedures. These procedures cannot be managed through Net-Net Central.

Net-Net Central Applications

Net-Net Central's core-applications include Device Manager, Element Manager, Health Service Monitor, and Security Manager, among others. With Device Manager you add and manage devices and device groups. The Dashboard Manager provides a summary view with at-a-glance device status and key performance indicators for your managed devices. With Security Manager, you control user-access and group-access to devices by setting and granting permissions.

Network element management core applications include Configuration Manager, where you load and provision your devices, Fault Manager, where you view events, and Performance Manager, where you view performance statistics collected from the Net-Net SBC, such as system, SNMP, IP, environmental, and so on.

With Net-Net Central Element Manager, you can:

- Add and manage devices and device groups
- View device status, memory usage, CPU usage, including key performance indicators and alarm counts from the dashboard summary view
- Load a configuration copy of the device, on-demand, to Net-Net Central for editing
- View your own modifications made to a device via a local configuration view, which details your configuration changes
- Customize your configuration by choosing from three distinct configuration view styles. Top-level elements are grouped by:
 - Default (also called, Tree-view style): Top-level elements are displayed logically, according to the type of configuration required
 - ACLI: Top-level elements are displayed as they would appear in the ACLI: by media-manager, session-router, system, or security
 - List: Top-level elements are displayed in an alphabetically-ordered list
- Conduct view-to-view navigation: Switch from one configuration view to another configuration view, with the content area automatically refreshing to the last element displayed from the previous view

Managed Devices

Net-Net Central's Element Manager application allows for the loading, configuring and managing of devices (Net-Net SBCs). This release of Net-Net Central supports the following devices:

- C-Series—Also known as the Net-Net 4000 series and Net-Net 3000 series. The Net-Net 4000 series contains two systems: the Net-Net 4250 and Net-Net 4500. The Net-Net 3000 series contains two systems: the Net-Net 3800 (Sku 3810) and Net-Net 3820. The C-Series software releases supported by this release are:
 - SC(X)6.1.0m6
 - SC(X)6.2.0 GA - SC(X)6.2.0m6
- D-Series—Also known as the Net-Net 9000 series, it contains one system: the Net-Net 9200. The D-Series software releases supported by this release are:
 - SD7.0.0m6
- E-Series—Also known as the Net-Net 2000 series, it contains one system: the Net-Net 2600.
 - NNOS-E 3.6 GA - current

Configuration Elements in this Guide

The devices operate on dynamic data models that vary depending upon the software image of the device. This guide is a superset of configurations from all series of SBCs and several software versions. The examples will change as this guide is updated for future releases.

The configurations presented in this guide may vary slightly in presentation to the configurations present on your Local Configuration Copy (LCC). The LCC is the local copy of the configuration of the loaded device, and is stored in the local NNC database. The LCC loaded on your system reflects the configuration elements required for the model of your NNC device.

For a comprehensive list of configuration elements and instructions that cannot be completed using Net-Net Central, please consult the *Net-Net Central Configuration Guide: ACLI View*, and the *Net-Net Central ACLI Reference Guide*. You can view the

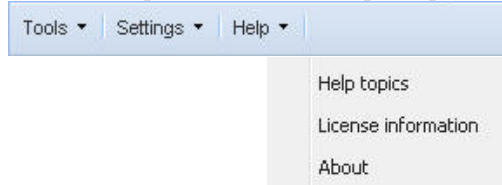
software version of your device in the Devices Element, as described below in the Device Data section.

Accessing Net-Net Central Version Information

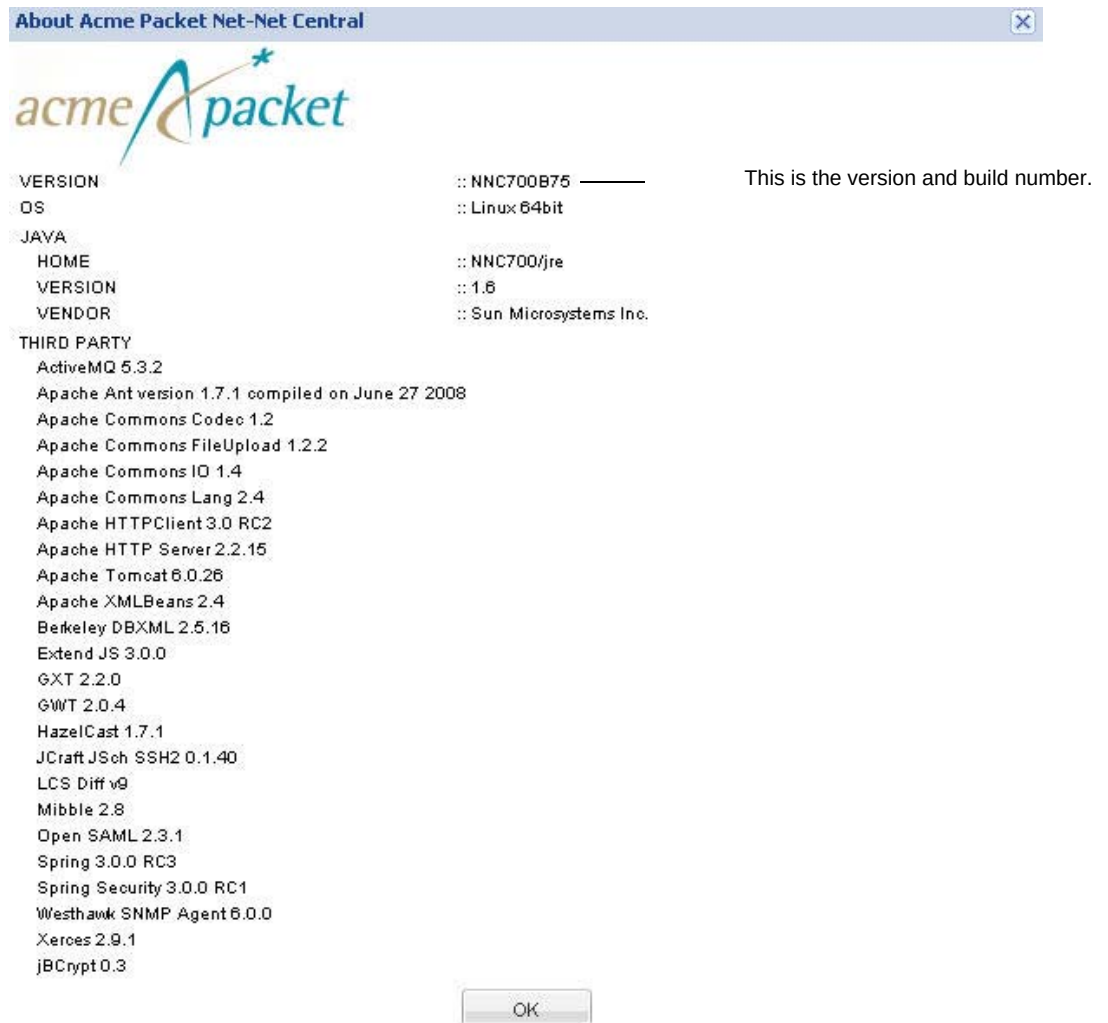
You can confirm the version of Net-Net Central you are using from the Help menu.

To access Net-Net Central version information:

1. From the top tool bar, click Help to open a drop down menu.



2. Click About. The About Acme Packet Net-Net Central dialog box appears.



3. Click OK to close this dialog box.

Identifying Net-Net SBC Versions

You can identify the specific type of Net-Net SBC you are managing by viewing the Managed Devices table under Device Manager, Devices option. The software version and hardware version are listed, as in the example below.

Managed Devices					
Device	Target Name	Health Score	Up Time	Software Version	Hardware Version
Home					
172.43.254.109	sd9	100	0Days 5:4:1	SCX620m3p6	3800
172.43.254.110	sd10	50	62Days 21:42:11	SCX620m3p5	4500
172.43.254.111	sd11	100	62Days 21:40:59	SCX620m3p5	4500
172.43.254.112	sd12	100	18Days 1:12:32	SCX620m3p5	4500

Net-Net Central Installation Requirements

This section explains how to install Net-Net Central in a Linux or Solaris operating system and how to migrate data from Net-Net EMS 6.x and Net-Net Route Manager Central 1.x.

System Requirements

Acme Packet has certified the following hardware and software server platforms; and client requirements for use with Net-Net Central 7.0.

Note: Other hardware configurations might work with Net-Net Central, but Acme Packet has verified the configurations listed here.

Solaris

- Netra x64/x86
- CPU: 4-core 2.1 GHz processor or better
- 12 GB RAM minimum
- 195 GB hard drive minimum, 300 GB hard drive recommended
- Solaris 10 64 bit with the following:
 - OpenSSL 0.9.7d 17 Mar 2004 (+ security fixes for: CVE-2005-2969 CVE-2006-2937 CVE-2006-2940 CVE-2006-3738 CVE-2006-4339 CVE-2006-4343 CVE-2007-5135 CVE-2008-5077 CVE-2009-0590)
- Solaris patches:
 - February 2010 recommended patches
 - SUNWcry and SUNWcryr for OpenSSL

Linux

- CPU: 4-core 2.1 GHz processor or better
- 12 GB RAM minimum
- 195 GB hard drive, 300 GB hard drive recommended
- Linux Red Hat Fedora Core 13 64 bit or Red Hat Enterprise Linux 5.5 64 bit

Network Manager with Fedora Core 13

Fedora Core 13 installs an application called Network Manager by default. Network Manager is used to configure network connections. It executes automatically when

you start your session and it is visible as an applet icon. You need to check your system to see if Fedora Core 13 installed Network Manager on your system. If installed you need to remove it and then turn on the network services.

To check for and remove Network Manager:

1. Check for Network Manager.
service NetworkManager status
2. Shutdown the Network Manager.
service NetworkManager stop
3. Remove Network Manager using the following command:
yum remove NetworkManager

To turn on network services:

chkconfig network on
service network start

Note: You should also ensure that the option Controlled by NetworkManager in Network Configuration is unchecked on your system.

OpenSSL

Most Solaris and Linux distributions include OpenSSL as part of the OS installation. For Solaris 10, OpenSSL 0.9.7d is installed by default. You can check the version on your system by using the following command:

openssl version
OpenSSL 0.9.7d Mar 27 2004

Client Requirements

- Internet Explorer versions 7.0 and higher (best performance with Internet Explorer 9.0) or Mozilla Firefox versions 3.0 and higher
- Flash player compatible with your browser installed locally
- If the server is not part of your DNS domain, the hosts file on each client must be edited to include the hostname and IP address of the NNC server. The client host file is usually located in the following directory:
windows\system32\drivers\etc

Using the DNS Database

All NNC servers and clients should be configured to use the DNS database for host name lookups. NNC servers should be defined in the DNS database.

If you are not using the DNS service, you must ensure the hosts file on all NNC servers and clients contain entries for the NNC server.

Note: If you want to connect to an NNC server over a Secure Sockets Layer (SSL) connection, you must have administrator privileges on the client system.

Information You Need

Ensure that you have identified the following information before you install:

- Hostname and IP address/netmask of the Net-Net Central server, as well as the IP addresses of its gateway, subnet mask, and DNS server
- IP address for each Net-Net SBC

- SNMP community strings for each Net-Net SBC

Net-Net Central Components Required

To install Net-Net Central, you need to obtain the appropriate tar.gz file for your environment from your Acme Packet representative.

- NNC700Linux64.tar.gz for Linux RHEL v5.5 64 bit installation
- NNC700FC1364.tar.gz for Linux Fedora Core 13 64 bit installation
- NNC700Solaris64x86.tar.gz for Solaris 64 bit x86 installation

You also need the AcmePacketNetNetCentral.xml license file.

Opening Ports on the Firewall

If you have a firewall placed between the NNC server and the Net-Net SBC or between the NNC server and the NNC clients, you must ensure the following ports are open.

Port Number	Protocol	Service	Configurable	Impacts Firewall	Purpose
Between NNC server and Net-Net SBC					
161	UDP	SNMP	N	Y	SNMP read/write requests between the NNC server and the Net-Net SBC.
162	UDP	SNMP	N	Y	SNMP trap reporting from the Net-Net SBC to the NNC server.
8080	HTTP		N	Y	Used for notifications to Net-Net SBC
Between NNC server and NNC clients					
1098	TCP/SSL	RMI	N	Y	SSL RMI
1099	TCP/SSL	RMI	N	Y	RMI registry port. Used for the RMI communication between the client and the server. SSL is used only if HTTPs setup is enabled.
5000		Hazelcast			Distributed data structures, peer-to-peer collective data distribution
8005	TCP	HTTP	N	Y	Tomcat shutdown port that is used by the shutdown script. Can be blocked on a firewall because it is local to the NNC server.
8009	TCP	Apache	N	Y	Tomcat port.
8443	TCP/SSL	HTTPS	N	Y	Apache port. HTTPS port for client/server communication.
8080	HTTP	HTTP	N	Y	Used for notifications to Net-Net SBC
9000			N	Y	Berkley database
61616			N	Y	Message broker

Either port 8080 (HTTP) or port 8443 (HTTPS) needs to be open on the firewall, depending on which one is chosen between the NNC client and server. If installing on a Linux system, the Linux firewall must also have either 8080 (HTTP) or port 8443 (HTTPS) open.

Net-Net Central Virtual Environment on Linux

This section contains information about obtaining, installing, and configuring your Linux system with a guest operating system that is part of using Net-Net Central in a virtual environment. Before following the information in this section, ensure you have contacted your Acme Packet representative about setting up the host operating system and installing and configuring VMware on your system.

You can install Net-Net Central as part of a virtual machine (VM) if you want to use a virtualized environment on your Linux system. You obtain a VM image containing the Net-Net Central application from the Acme Packet customer portal site.

You can use Net-Net Central VM with the VMware server. VMware transforms (virtualizes) your system's hardware resources—including the CPU, RAM, hard disk and network controller—to create a fully functional virtual machine.

You can use either of the following VMware products:

- Commercially available versions such as VMware ESX
- Freeware products such as VMware Player, VMware Server, and VMware ESXi

Note: Net-Net Central VM will not support Net-Net Central clustering.

Before You Start

Review the following information before proceeding to the instructions. You need to set up your host operating system and install VMware first and have information ready when configuring.

VMware Server Setup

Contact your Acme Packet representative about setting up the host operating system and installing and configuring VMware server or ESXi installed on your system before proceeding with installing the guest operating system and the Net-Net Central virtual machine.

VMware Player

You need to install the player software on the host machine.

About the Net-Net NNC VM Image

The Net-Net NNC VM image includes the following two components:

- VM guest operating system (VM's own operating system)
- VM-NNC700FC1364bit.tar.gz

Memory Usage

Do not set more virtual memory than you have in physical memory.

Check /etc/hosts definition order

Net-Net Central requires that the host machine's IP address is properly set. In order to avoid the localhost IP address 127.0.0.1 being used during a cluster setup, it is important to note the order of the IP address defined in the /etc/hosts file.

1. Login as root or su to root
2. Verify the order in the /etc/hosts file follows the following format:

<Host IP Address> <Host name> <Host domain name>

127.0.0.1 localhost

```
::1          localhost
```

For example:

```
172.30.200.88  acmefc1364vm  acmefc1364vm.acmepacket.com
127.0.0.1      localhost
::1           localhost
```

Installing VM NNC

To install VM NNC:

1. Copy the VMWare image to the server running any of the following:
 - VMWare Server
 - VMWare ESX
 - VMWare ESXi
2. Unzip the *.tar.gz file.
gunzip <filename>.tar.gz
3. Extract the file(s):
tar -xvf <filename>.tar
 The file NNC700-opt-0.vmdk is the second hard drive for the VMWare image, mounted as /opt inside the FC 13 64 bit.
4. If running VMWare ESX or ESXi, add it to the inventory. If running VMWare server, add it to the virtual machines directory.
5. Import the virtual machine to your VMWare server.
6. Start the Net-Net Central virtual machine.
7. Login using any one of the follow username; password sets.
 - guest; guest123
 - nncentral; nncentral123
 - root: root123
8. If you login as guest or nncentral, run the su command to change to root user.
9. Configure the Ethernet interface to use the IP address for your network.
10. Edit the /etc/host/ file to use the same network IP address.
11. Change directory to /opt/AcmePacket/NNC700/bin.
12. Start Net-Net Central using the ./startnnc.sh command.

Before a New Installation

This section explains how to configure your operating system before you install Net-Net Central for the first time. You should contact your Acme Packet systems engineer for a copy of the latest NNC Best Current Practices document. It contains instructions on how to install the Linux and Solaris operating systems.

Configuring Linux

To configure Linux:

- Include the Linux host name
- Disable the default http daemon

Including the Linux Hostname

You must configure the Linux system hostname during the installation of the operating system. You can determine the hostname by using the hostname command on the Linux system. For example:

```
[bash]$ hostname
```

```
nncsvr
```

You need to edit the /etc/hosts file to include the Linux system hostname in the following format:

```
<IP address>    <hostname>    <hostname>.local domain
```

The following example shows the inclusion of a server named nncsvr with an IP address of 10.0.0.252:

```
[bash]$ cat /etc/hosts
```

```
#Do not remove the following line, or various programs
```

```
# that require network functionality will fail.
```

```
127.0.0.1        localhost        localhost.localdomain
10.0.0.252       nncsvr           nncsvr.localdomain
```

Disabling the Default HTTP Daemon

You need to disable the default http daemon process on the NNC server.

To disable the http daemon:

1. Log in as root user and open a Terminal window.
2. From the prompt, run the setup command.
The Choose a Tool screen displaying the type of services appears.
3. Select system services to refresh the list of system services available/running on the system.
4. If listed, deselect the https service.
5. Save the configuration.
6. Reboot the system. The http daemon process should be disabled.
7. Verify the daemon is disabled by making an HTTP query.

OpenSSL Required for HTTPS

If you plan to use HTTPS, you need OpenSSL installed on the Net-Net Central server before you install Net-Net Central. Solaris 10 servers should have OpenSSL already installed. Check the version using the following command:

```
openssl version
```

Downloading OpenSSL

If not already installed, download the following OpenSSL file from www.openssl.org/source/:

```
openssl-0.9.7e.tar.gz
```

Follow the directions supplied by OpenSSL to install the files.

Displaying Shared Libraries

You need to display the shared libraries and check that they are connected. If any libraries are not connected, you can create soft links for them. For example, in Fedora Core 13 you might need to create soft links for two libraries.

To display shared libraries:

1. As root user, change to the httpserver bin directory. For example:

```
cd /opt/AcmePacket/NNC700/Apache/httpserver/bin
```

2. Run the following command to display the shared libraries:

```
ldd httpd
```

Output similar to the following for Fedora Core 13 appears:

```
linux-vdso.so.1 => (0x00007fff9e8b0000)
libm.so.6 => /lib64/libm.so.6 (0x0000003b7f400000)
libaprutil-1.so.0 => /usr/lib64/libaprutil-1.so.0 (0x00007f85607ea000)
libexpat.so.0 => (file not found)
libuuid.so.1 => /lib64/libuuid.so.1 (0x0000003b83800000)
librt.so.1 => /lib64/librt.so.1 (0x0000003b80400000)
libcrypt.so.1 => /lib64/libcrypt.so.1 (0x0000003b8fe00000)
libpthread.so.0 => /lib64/libpthread.so.0 (0x0000003b7fc00000)
libdl.so.2 => /lib64/libdl.so.2 (0x0000003b7f800000)
libc.so.6 => /lib64/libc.so.6 (0x0000003b7f000000)
libdb-4.8.so => /lib64/libdb-4.8.so (0x0000003b95200000)
/lib64/ld-linux-x86-64.so.2 (0x0000003b7e800000)
libfreebl3.so => /lib64/libfreebl3.so (0x0000003b90200000)
```

One of the shared libraries libexpat.so.0 is not found. A soft link must be created for it.

Creating Soft Links

You can create soft links for any shared libraries that are not connected.

To create soft links:

1. As root user, change directory to /usr/lib64.
2. Create links for any unlinked shared libraries. For example:

```
ln -s libexpat.so.1.5.2 libexpat.so.0
```

```
ln -s libexpat.so.1.5.2 ../lib64/libexpat.so.0
```

Creating nncentral Group and User

For security reasons, you can create an account named `nncentral` and a group named `nncentral`. You also must install the UNIX sudo facility and define sudo privileges. After the Net-Net Central installation, all the installed files are owned by `nncentral`. The main Net-Net Central process has to run as sudo user in order to have access to port 162.

Linux

To create nncentral group and user:

1. Login as root.
2. Enter the following lines:

```
groupadd nncentral
useradd -m -g nncentral -d /home/nncentral -s /usr/bin/bash nncentral
passwd nncentral
```

Solaris

To create nncentral group and user:

1. Login as root.
2. Enter the following lines:

```
groupadd nncentral
useradd -m -g nncentral -d /export/home/nncentral -s /usr/bin/bash nncentral
passwd nncentral
```

Defining sudo Privileges

You must have the UNIX sudo facility installed on the server for deployments with user `nncentral` so that you can grant sudo privilege to the `nncentral` user.

The sudo facility comes by default with Linux, but not on Solaris 10. If using the Solaris, download the utility from the Sun site. For example:

<http://www.sudo.ws/sudo/>

Installing sudo

To install sudo:

1. Login as root.
2. Copy the binary package to a directory. For example:

```
cp TCMsudo-1.7.4p2-i386.pkg.gz /tmp/.
```
3. Go to the directory the sudo binary package resides in. For example:

```
cd /tmp
```
4. Decompress the package.

Note: Depending on advertised mime type/encoding, most browsers will automatically decompress a gzipped file. So although your files end in `.gz`, they might actually be already decompressed.
5. Add the decompressed sudo package to the system. For example:

```
pkgadd -d TCMsudo-1.7.4p2-i386.pkg.gz
```

The following packages are available:

 - 1 TCMsudo
 - sudo 1.7.4p2
 - (amd64) 1.7.4p2
6. Select the package(s) you want to process or 'all' to process all packages.

Processing package instance <TCMsudo> from </tmp/TCMsudo-1.7.4p2-i386.pkg.gz>

sudo 1.7.4p2(amd64) 1.7.4p2

(c) 1993-1996,1998-2010 Todd C. Miller

Processing package information.

Processing system information.

Verifying disk space requirements.

Checking for conflicts with packages already installed.

Checking for setuid/setgid programs.

The following files are being installed with setuid and/or setgid permissions:

/usr/local/bin/sudo <setuid root>

/usr/local/bin/sudoedit <setuid root>

The following prompt appears:

Do you want to install these as setuid/setgid files [y,n,?,q]

7. Enter y and press Enter. This package contains scripts which will be executed with super-user permission during the process of installing this package. The following prompt appears:

Do you want to continue with the installation of <TCMsudo> [y,n,?]

8. Enter y and press Enter.

Installing sudo 1.7.4p2 as <TCMsudo>

Installing part 1 of 1.

/etc/sudoers.dist

/usr/local/bin/sudo

/usr/local/bin/sudoedit

/usr/local/bin/sudoreplay

/usr/local/libexec/sudo_noexec.so

/usr/local/libexec <implied directory>

/usr/local/sbin/visudo

/usr/local/sbin <implied directory>

/usr/local/share/doc/sudo/ChangeLog

/usr/local/share/doc/sudo/HISTORY

/usr/local/share/doc/sudo/LICENSE

/usr/local/share/doc/sudo/NEWS

/usr/local/share/doc/sudo/README

/usr/local/share/doc/sudo/TROUBLESHOOTING

/usr/local/share/doc/sudo/UPGRADE

/usr/local/share/doc/sudo/sample.pam

/usr/local/share/doc/sudo/sample.sudoers

/usr/local/share/doc/sudo/sample.syslog.conf

/usr/local/share/man/man1m/sudo.1m

/usr/local/share/man <implied directory>

/usr/local/share/man/man1m <implied directory>

/usr/local/share/man/man1m/sudoedit.1m

/usr/local/share/man/man1m/sudoreplay.1m

/usr/local/share/man/man1m/visudo.1m

/usr/local/share/man/man4/sudoers.4


```

/usr/local/share/man/man4 <implied directory>
[ verifying class <run> ]
## Executing postinstall script.
Installation of <TCMsudo> was successful.

```

Editing the sudoer Configuration File

You need to use the visudo tool to make edits to the sudoer configuration. The visudo tool is based on vi editor; you can use all vi editor commands. The Net-Net Central administrator needs to provide a sudo password when starting Net-Net Central.

1. Login as root.
2. Change directory to visudo. For example:
cd /usr/bin/visudo
3. Add the following lines to the sudoer configuration using vi commands:

```

root <DNS hostname>=(ALL)ALL
Defaults:ALL timestamp_timeout=0
nncentral <DNS hostname>=(ALL)ALL

```

The following example shows the lines added using the hostname **mynnchost**:

```

root mynnchost=(ALL)ALL
Defaults:ALL timestamp_timeout=0
nncentral mynnchost=(ALL)ALL

```

Installing Net-Net Central

This section explains how to install and setup Net-Net Central on the NNC server. The steps are the same for both the Linux and Solaris operating systems.

To install Net-Net Central, you need to obtain the appropriate tar.gz file for your environment and the AcmePacketNetNetCentral.xml license file from your Acme Packet representative. You explode the tar file contents on your server and run the setup script to configure Net-Net Central.

Installing Net-Net Central

To install Net-Net Central:

1. Obtain the appropriate tar.gz file from the Acme Packet customer portal. The tar.gz files include the following:
 - NNC700Linux64.tar.gz for Linux RHEL v5.5 64 bit installation
 - NNC700FC1364.tar.gz for Linux Fedora Core 13 64 bit installation
 - NNCSolaris64x86.tar.gz for Solaris 64 bit x86 installation
2. FTP the tar.gz file to your system. Place the file in the directory where you want the Net-Net Central software to reside after you unzip the tar.gz file. For example, FTP the tar file to the /opt directory.
3. Unzip the *.tar.gz file.
gunzip <filename>.tar.gz
4. Extract the files using one of the following commands:
 - Linux
tar -xvf <filename>.tar
 - Solaris
/usr/sfw/bin/gtar -xvf <filename>.tar

From here you run setup to configure Net-Net Central.

Running Setup

The following steps show running setup on a Linux system. The process is the same for Solaris.

To run setup:

1. Login as root user.
2. Navigate to the bin directory. For example:
cd /opt/AcmePacket/NNC700/bin
3. Run setup.sh.
./setup.sh

A Welcome message appears and initialization processes occur. Setup checks that minimal system requirements are met and checks system port availability for NNC components.

```

/apps/AcmePacket/NNC700B68/bin
[root@pixie bin]# ./setup.sh

=====

Welcome to NNC Setup application
Version : NNC700B68
OS : Linux : amd64 : 2.6.34.7-61.fc13.x86_64

=====

Please wait while application loads
Checking environment and setting permissions.
Please wait ....
100%[=====]

=====

System Physical Memory Diagnostics
Total System Physical Memory = 7900 MB
Total System Free Physical Memory = 7332 MB
WARNING: Free physical memory is insufficient for running this
application.
The recommended total required memory that should be available is =
12000 MB
Dynamic memory allocation in progress
Previous database cache 764411904
New database cache 764411904
Previous JVM Xmx size 2918
New JVM Xmx size 2918

=====

System Disk Space Diagnostics
Total System Disk Space = 389 GB
Free System Disk Space = 384 GB

=====

System Port Availability Diagnostics
The following port is available [ 5000 ]
The following port is available [ 8080 ]
The following port is available [ 61616 ]
The following port is available [ 9000 ]
The following port is available [ 8443 ]
The following port is available [ 1099 ]
The following port is available [ 8009 ]
The following port is available [ 1098 ]
The following port is available [ 8005 ]

=====

Note: Disregard the warning message about free physical memory if
you have one.

```

After the diagnostics, the setup options appear.

Set up options

```

TYPICAL      : This setup procedure walks the user through
                the minimal setup configuration required to
                configure Net-Net server.

CUSTOM       : This setup procedure provides the user with
                a set of options to manually pick and
                choose from.

[X] 1 - Typical    : Runs through most common set up options. (Recommended) [Default]
[ ] 2 - Custom     : Allows manual customization. (Advanced users)
[ ] 3 - Quit       : Finish and quit setup.
  
```

Please select an option [1] █

Choose the Typical setup option if you are new to Net-Net Central or just want to create the minimal configuration required by the NNC server. Choose Custom if you are a more advanced user or are licensed for applications hosted on Net-Net Central. The Custom setup also includes the options available on the Typical setup.

You can run the setup repeatedly to changed existing configuration values and to access any new Custom options as a result of licensing additional applications.

Typical Setup

The Typical setup creates the minimal configuration required by the NNC server in order to run properly. If you are new to Net-Net Central, you might want to start with the Typical setup.

To perform a typical setup:

1. Press Enter to accept the default value 1 for a typical setup. You are prompted about continuing.
2. Enter Y and press Enter to continue. The list of Typical configuration options appears.

```
Do you want to continue Yes/No?Y
```

```
=====
```

```
Typical Configuration
Will walk through basic configuration options.
```

```

[X] 1 - Check and Apply License    [Default]
[ ] 2 - HTTP/HTTPS configuration
[ ] 3 - Configure Sudo user password
[ ] 4 - Quit setup
  
```

Please select an option [1] █

3. Press Enter to accept the default value of 1 for checking and applying the Net-Net Central license. You are prompted about continuing.

4. Enter Y and press Enter to continue. The license options appear.

```
Do you want to continue Yes/No?Y
```

```
=====
```

```
Check and Apply License
```

```
This option allows the end user to validate an existing license,  
provide a license or re-apply a Acme Packet Net-Net Central license
```

```
Acme Packet Net-Net Central management
```

```
Acme Packet Net-Net Central license will be checked, applied and validated.
```

```
Please choose to quit license management or apply new license
```

```
[X] 1 - Quit [Default]
```

```
[ ] 2 - Apply new license
```

```
Please select an option [1] █
```

Applying the License

You need a valid license to be able to start the NNC servers. You can apply a new license or check an existing license and accept that.

To apply the license:

1. Enter 2 and press Enter. You are prompted about continuing.
2. Enter Y and press Enter to continue. You are prompted for the license directory and filename.

```
Do you want to continue Yes/No?Y
```

```
Please specify the source directory and file name of Acme Packet Net-Net license
```

```
Source Directory [      ] █
```

3. Enter the name of the directory where the license file is located and the name of the license file and press Enter. For example:

```
/opt/AcmePacketNetNetCentral.xml
```

The license is located and the details appear on the screen. You are prompted about accepting the license.

```
Please specify the source directory and file name of Acme Packet Net-Net license
Source Directory [          ] /opt/AcmePacketNetNetCentral.xml
Source Directory [/opt/AcmePacketNetNetCentral.xml]

License Details
License number      : -1002
ClientName          : AcmePacket-Internal
LicenseType         : permanent
IssuedDate          : Thu Sep 30 11:43:00 EDT 2010
Key                 : 2.0.1
StartDate           : 2010-09-30
ExpirationDate      :
Cluster             : Enabled

Applications
Application          Number of licensed devices
RouteManager         : 8000

Accepting this license will replace an existing license
[X] 1 - Accept license   [Default]
[ ] 2 - Do not accept license

Please select an option [1] █
```

4. Press Enter to accept the default and accept the license. You are prompted about continuing.
5. Enter Y and press Enter to continue. The license is accepted and the list of setup options appears.

```
Do you want to continue Yes/No?Y
```

```
[ ] 1 - Check and Apply License   [Default]
[X] 2 - HTTP/HTTPS configuration
[ ] 3 - Configure Sudo user password
[ ] 4 - Quit setup
```

```
Please select an option [2] █
```

The next setup option is for configuring the Apache load balancer to run in either HTTP or HTTPS mode. The Apache load balancer is responsible for distributing the load by redirecting traffic to the front end servers.

HTTP**To configure HTTP mode:**

Note: You cannot use the value root for either the user or group name.

1. Press Enter to accept the default. You are prompted about continuing.
2. Enter Y and press Enter to continue. The HTTP/HTTPS options appear.

Do you want to continue Yes/No?Y

=====

HTTP/HTTPS configuration

Set up HTTP or HTTPS configuration

```
[X] 1 - HTTP mode - Configure server to run in HTTP mode    [Default]
[ ] 2 - HTTPS mode - Configure server to run in HTTPS mode
```

Please select an option [1] █

3. Press Enter to accept the default to configure HTTP. You are prompted about continuing.
4. Enter Y and press Enter to continue. You are prompted for the username of the server process.

Do you want to continue Yes/No?Y

HTTP mode

Configure server to run in HTTP mode

Enter the user name of the server process which determines what files the server is allowed to access. Any files inaccessible to this user are also inaccessible to clients connecting to the Apache HTTP Server.

Apache User [nncentral] █

5. Press Enter to accept the default value nncentral. You are prompted for the group name.

Enter the group name of the Apache HTTP Server processes

Apache Group [nncentral] █

6. Press Enter to accept the default value nncentral. You are prompted for the port number.

Enter the port number that the Apache HTTP Server should listen on

Apache Port Number (1024-65535) [8080] █

7. Press Enter to accept the default value 8080. The list of Typical setup options appears.

Apache Port Number (1024-65535) [8080]

```
[ ] 1 - Check and Apply License    [Default]
[ ] 2 - HTTP/HTTPS configuration
[X] 3 - Configure Sudo user password
[ ] 4 - Quit setup
```

Please select an option [3] █

Note: You cannot use a port number reserved for Net-Net Central components.

HTTPS**To configure HTTPS mode:**

Note: You cannot use the value root for either the user or group name.

1. Enter 2 to configure HTTPS and press Enter. You are prompted about continuing.

```
Please select an option [1] 2
```

```
[ ] 1 - HTTP mode - Configure server to run in HTTP mode   [Default]
[X] 2 - HTTPS mode - Configure server to run in HTTPS mode
```

```
Do you want to continue Yes/No? █
```

2. Enter Y and press Enter to continue. You are prompted for the user name for the server process.

```
Do you want to continue Yes/No?Y
```

```
HTTPS mode
Configure server to run in HTTPS mode
```

```
Enter the user name of the server process which determines what files the server is allowed to access. Any files inaccessible to this user are also inaccessible to clients connecting to the Apache HTTP Server.
```

```
Apache User [nncentral] █
```

3. Press Enter to retain the default nncentral. You are prompted for the group name.

```
Apache User [nncentral]
```

```
Enter the group name of the Apache HTTP Server processes
Apache Group [nncentral] █
```

4. Press Enter to retain the default nncentral. You are prompted for the port number the server should listen on.

```
Enter the port number that the Apache HTTP Server should listen on
Apache Port Number (1024-65535) [8443] █
```

Note: You cannot use a port number reserved for Net-Net Central components.

5. Press Enter to retain the default 8443. You are prompted for the server's DNS name.

```
The server name(DNS name of this server)
Server name [] █
```

6. Enter the DNS name of the server and press Enter. You are prompted about creating a self-signed certificate.

```
Would you like to create a self signed certificate?
[ ] 1 - Yes
[X] 2 - No   [Default]
```

```
Please select an option [2] █
```

Skip to Creating Self-Signed Certificates to create a self-signed certificate.

No Self-Signed Certificate

1. Retain the default value No if you do not want to create a self-signed certificate.
2. Press Enter. You are prompted about continuing.
3. Enter Y to continue and press Enter. You are prompted for the private key file.

Do you want to continue Yes/No?Y

The private key file
Private key file [] █

4. Enter the file name, including the path, and press Enter. You are prompted for the certificate file.
5. Enter the file name, including the path, and click Enter. You are prompted about intermediate certificates.
6. Press Enter to accept the default. You are prompted about continuing.
7. Enter Yes to continue. You are prompted for the alias name for the certificate in the truststore.
8. Press Enter to accept the default or enter a different alias name and press Enter. You are prompted for the trustore password.
9. Press Enter to accept the default or enter a different password and press Enter.

Creating Self-Signed Certificate**To create a self-signed certificate:**

1. Enter 1 if you want to create a self-signed certificate and press Enter. You are prompted about continuing.

Please select an option [2] 1

[X] 1 - Yes
[] 2 - No [Default]

Do you want to continue Yes/No?Y

2. Enter Y to continue and press Enter. You are prompted for the common name (DNS name) of the server.
3. Enter the DNS name of the server and press Enter. You are prompted for the certificate alias name in the trustore.
4. Retain the default or enter a different alias name and press Enter. You are prompted for the trustore password.
5. Enter the trustore password and press Enter. The Typical install list of options appears.

[] 1 - Check and Apply License [Default]
[] 2 - HTTP/HTTPS configuration
[X] 3 - Configure Sudo user password
[] 4 - Quit setup

Please select an option [3] █

From here you configure the sudo password. Net-Net Central uses the sudo password to support internal components that require sudo privileges.

Note: Verify with your system administrator that you have the correct sudo password before continuing.

Sudo Password

Net-Net Central requires the sudo password to run programs with certain privileges. You set the sudo password to match the sudo password that has been defined by the system administrator.

To configure a sudo password:

1. Press Enter to accept the default. You are prompted about continuing.
2. Enter Y and press Enter. Two options appear.

```
Do you want to continue Yes/No?Y
```

```
=====
```

```
Configure Sudo user password
```

```
Provides the ability to set the Sudo user password in NNC to the  
Password that an administrator has already defined.
```

```
Net-Net Central requires entry of the sudo password in order to  
support internal components that require sudo user privileges.  
The password you supply will be securely encrypted.
```

```
[X] 1 - Enter sudo password [Default]  
[ ] 2 - Return to main menu
```

```
Please select an option [1]
```

3. Press Enter to accept the default value Enter sudo password. You are prompted about continuing.
4. Enter Y and press Enter. You are prompted for the sudo password.

```
Do you want to continue Yes/No?Y
```

```
Enter sudo password: []
```

5. Enter your user password and press Enter. For example, nncentral. You are prompted to confirm the password.
6. Enter the password again and press Enter. The confirmation message appears.

```
Confirm sudo password: []  
Sudo password entered and encrypted successfully!
```

```
[ ] 1 - Check and Apply License [Default]  
[ ] 2 - HTTP/HTTPS configuration  
[ ] 3 - Configure Sudo user password  
[X] 4 - Quit setup
```

```
Please select an option [4]
```

7. Press Enter to accept the default Quit setup. You are prompted about continuing.
 8. Enter Y to continue and press Enter. You exit the setup program.
- You can continue to install custom options by running the setup.sh script again and choosing the Custom option.

Custom Setup

Custom setup options are for the more advanced user or for those users who are licensed for separate applications. The setup options for those applications are found are included in Custom. For example, if you are licensed for Route Manager, For example, if you are licensed for Route Manager, options for Route Manager Central and SAML Single sign on appear in the list.

What appears on the list of setup options can differ depending on what applications you are licensed for. One setup option for the Mail Server always appears on the list but is only applicable if you have a license for Element Manager. This section includes the instructions for all currently supported applications. What you see can be different depending on your licenses.

You can run the setup.sh script repeatedly to access the Custom setup options.

Accessing Custom Options

The following instructions are based on configuring setup options on a Linux system. The steps are exactly the same for a Solaris system.

To access custom setup options:

1. Login as root user.
2. Navigate to the bin directory. For example:
3. Run setup.sh.

```
cd /opt/AcmePacket/NNC700/bin
```

```
./setup.sh
```

A Welcome message appears and initialization processes occur. Setup checks that minimal system requirements are met and checks system port availability for NNC components.

Note: Disregard the warning message about free physical memory if you have one.

After the diagnostics, the setup options appear.

Set up options

```
TYPICAL      : This setup procedure walks the user through
                the minimal setup configuration required to
                configure Net-Net server.

CUSTOM       : This setup procedure provides the user with
                a set of options to manually pick and
                choose from.

[X] 1 - Typical    : Runs through most common set up options. (Recommended) [Default]
[ ] 2 - Custom     : Allows manual customization. (Advanced users)
[ ] 3 - Quit       : Finish and quit setup.
```

Please select an option [1] █

4. Enter 2 for Custom and press Enter. You are prompted about continuing.
5. Enter Y and press Enter. The custom setup options appear.

```
Do you want to continue Yes/No?Y
```

```
=====
```

```
Customize Configuration
Will walk through advanced configuration options.
```

```
[X] 1 - Check and Apply License [Default]
[ ] 2 - HTTP/HTTPS configuration
[ ] 3 - Configure Sudo user password
[ ] 4 - Net-Net Central cluster management.
[ ] 5 - Route Manager Central configuration
[ ] 6 - SAML Single sign on configuration
[ ] 7 - Mail Server configuration
[ ] 8 - Quit setup
```

Please select an option [1] █

See the information in Typical Setup for details about the following options:

- Check and Apply License
- HTTP/HTTPS configuration
- Configure Sudo user password

See the following sections for details about the custom options.

Cluster Management

If you are licensed for Element Manager, the Net-Net Central cluster management option appears. Access this option to configure and manage a cluster of NNC servers for High Availability (HA). See the *Net-Net Central High Availability Guide* for more information about HA and clusters.

Note: Ensure you synchronize time on each server with NTP before adding it to a cluster.

To configure clusters:

1. Enter 4 and press Enter. You are prompted about continuing.
2. Enter Y and press Enter. The cluster management options appear.

Do you want to continue Yes/No?Y

=====

Net-Net Central cluster management.

This option is used to configure NNC as part of a cluster

Management options

The host machine can run as a standalone or member of a Net-Net Central cluster.

Please choose to quit cluster management or apply new configuration

[X] 1 - Configure and manage members in cluster [Default]

[] 2 - Run current host as a standalone

[] 3 - Quit out of cluster configuration

Please select an option [1] █

Manage Cluster Members

1. Press Enter to configure and manage members in a cluster. You are prompted about continuing.
2. Enter Yes and press Enter. The cluster member management options appear.

Do you want to continue Yes/No?Yes

Host name	DB	MOM	Web Worker
	Port	Port	Port
localhost	9000	61616	8009

Use the following options to add or remove members from cluster.

[] 1 - Add a new member

[] 2 - Remove all remote members

[] 3 - Apply new cluster configuration

[X] 4 - Cancel out and do not apply changes [Default]

Please select an option [4] █

Adding New Members

To add new member to a cluster:

1. Enter 1 and press Enter. You are prompted about continuing.
2. Enter Y and press Enter. You are prompted for the DNS name or IP address of the host you want to add.

Do you want to continue Yes/No?Y

Host name	DB	MOM	Web Worker
	Port	Port	Port
localhost	9000	61616	8009

Provide the DNS name or IP address of the Host requiring membership to cluster.

Member host name [] █

3. Enter the DNS name or the IP address for the host you are adding to the cluster and press Enter. A confirmation message appears and the options appear again.

```

Provide the DNS name or IP address of the Host requiring membership to cluster.
Member host name [ ] 172.30.80.19
Valid remote member has been added
Host name          | DB      | MOM      | Web Worker
                   | Port    | Port     | Port
localhost          | 9000    | 61616    | 8009

===== Remote members =====
172.1.30.185       | 9000    | 61616    | 8009
172.30.80.19       | 9000    | 61616    | 8009

Use the following options to add or remove members from cluster.
[ ] 1 - Add a new member
[ ] 2 - Remove all remote members
[X] 3 - Apply new cluster configuration
[ ] 4 - Cancel out and do not apply changes [Default]

Please select an option [3] 

```

4. Repeat steps to add additional hosts to the cluster. When you are done adding hosts to the cluster you can apply the new cluster configuration.
5. Press Enter to accept the default 3 Apply new cluster configuration. (Or you can cancel out of creating a cluster without applying your changes.) You are prompted about continuing.
6. Enter Yes and press Enter. The cluster members are displayed and the cluster management options appear.

```

Do you want to continue Yes/No?Y
Host name          | DB      | MOM      | Web Worker
                   | Port    | Port     | Port
172.30.1.185       | 9000    | 61616    | 8009

===== Remote members =====
172.30.80.19       | 9000    | 61616    | 8009
172.30.80.9        | 9000    | 61616    | 8009
172.30.80.198      | 9000    | 61616    | 8009

Please choose to quit cluster management or apply new configuration
[X] 1 - Configure and manage members in cluster [Default]
[ ] 2 - Run current host as a standalone
[ ] 3 - Quit out of cluster configuration

Please select an option [1] 

```

7. Enter 3 and press Enter to quit out of the cluster configuration. You are prompted about continuing.

8. Enter Y and press Enter. The Configure sftp information options appear and you are prompted about your host being a member of a Net-Net Central cluster.

```
Please select an option [1] 3
```

```
[ ] 1 - Configure and manage members in cluster   [Default]
[ ] 2 - Run current host as a standalone
[X] 3 - Quit out of cluster configuration
```

```
Do you want to continue Yes/No?Y
```

```
Configure sftp information
```

```
Make sure to configure sftp properties if there are members in the cluster
```

```
Will this machine be a member of a Net-Net Central cluster?
```

```
[ ] 1 - Yes
[X] 2 - No   [Default]
```

```
Please select an option [2] [ ]
```

9. Press Enter to retain the default value No or enter 1 to add your current host to the cluster. See the following section for instructions.

Adding Current Host to Cluster

You can include the host on which you are running setup in the cluster.

1. Enter Y to continue when prompted. A prompt appears for the username to use to SFTP files from the host.

```
Please select an option [2] 1
```

```
[X] 1 - Yes
[ ] 2 - No   [Default]
```

```
Do you want to continue Yes/No?Y
```

```
Please enter the username to use to sftp files off of this machine
Username [ ] [ ]
```

2. Enter the username you want to use to SFTP files and press Enter. You are prompted for the password.

3. Enter the password to use to SFTP files and press Enter. The Custom setup options appear.

```
Please enter the username to use to sftp files off of this machine
Username [] nncentral
```

```
Please enter the password for the username
Password []
```

```
[ ] 1 - Check and Apply License    [Default]
[ ] 2 - HTTP/HTTPS configuration
[ ] 3 - Configure Sudo user password
[ ] 4 - Net-Net Central cluster management.
[X] 5 - Route Manager Central configuration
[ ] 6 - SAML Single sign on configuration
[ ] 7 - Mail Server configuration
[ ] 8 - Quit setup
```

```
Please select an option [5] []
```

Removing Members from Cluster

There are two reasons why you would remove all members from a cluster. Either you are eliminating the cluster or you want to retain the cluster but remove a member. Both of these require you remove all members from the cluster. If you are eliminating the cluster, you are done. If you are removing a member, you need to re-add the members you want to retain to the cluster.

To remove members from the cluster:

1. Enter 4 and press Enter to choose Net-Net Central cluster management. You are prompted about continuing.
2. Enter Y and press Enter to continue. The cluster management options appear.
3. Press Enter to retain the default value. You are prompted about continuing.
4. Enter Y and press Enter to continue. The add and remove members options appear.

```
Do you want to continue Yes/No?Y
```

Host name	DB Port	MOM Port	Web Worker Port
172.30.1.185	9000	61616	8009

```
===== Remote members =====
```

172.1.30.185	9000	61616	8009
172.30.80.19	9000	61616	8009
172.30.80.9	9000	61616	8009

```
Use the following options to add or remove members from cluster.
```

```
[ ] 1 - Add a new member
[ ] 2 - Remove all remote members
[ ] 3 - Apply new cluster configuration
[X] 4 - Cancel out and do not apply changes    [Default]
```

```
Please select an option [4] []
```

5. Enter 2 to choose Remove all remote members and press Enter. You are prompted about continuing.

6. Enter Y to continue and press Enter. You are prompted that the operation removes all remote members from the edition configuration session. By default, the Cancel out of clear operation option is selected.

Do you want to continue Yes/No?Y

This operation will remove all remote members from the edition configuration session.
 [] 1 - Proceed with removing all remote members
 [X] 2 - Cancel out of clear operation [Default]

Please select an option [2] 1

7. Enter 1 to choose Proceed with removing all remote members. You are prompted about continuing.
8. Enter Y and press Enter to continue. All remote nodes are removed and only the host you on which you are running Net-Net Central is visible. The cluster management options also appear.

Do you want to continue Yes/No?Y

Host name	DB	MOM	Web Worker
	Port	Port	Port
172.30.1.185	9000	61616	8009

Use the following options to add or remove members from cluster.

[] 1 - Add a new member
 [] 2 - Remove all remote members
 [] 3 - Apply new cluster configuration
 [X] 4 - Cancel out and do not apply changes [Default]

Please select an option [4] []

9. Enter 3 to apply the new cluster configuration to apply your changes. You are prompted about continuing.
10. Enter Y and press Enter to continue.

Run Current Server as Standalone

You can configure the current server as a standalone to remove it from a cluster. All other members of the cluster are removed as well. You need to re-add any members you want to retain in the cluster.

1. Enter 2 and press Enter. You are prompted about continuing.
2. Enter Y and press Enter to continue. The cluster member list appears with a warning that the server is part of a cluster and configuring it as a standalone removes it from the cluster. (All other members are removed as well and need to be re-added to the cluster if you want to retain them.)
3. Press Enter to continue configuring the server as a standalone. You are prompted about continuing.

4. Enter Y to continue and press Enter. All members of the cluster are removed and the current server is now considered a standalone.

```

Do you want to continue Yes/No?Y
Host name                | DB      | MOM      | Web Worker
                        | Port    | Port     | Port
localhost                | 9000    | 61616    | 8009

Please choose to quit cluster management or apply new configuration
[X] 1 - Configure and manage members in cluster   [Default]
[ ] 2 - Run current host as a standalone
[ ] 3 - Quit out of cluster configuration

Please select an option [1] █

```

5. Press Enter to configure and manage members in cluster if you want to retain the other members. You need to re-add them to recreate the cluster. Or enter 3 and press Enter to quit the cluster configuration.
6. Enter Y to continue and press Enter.

Route Management Central

If you are licensed for the Route Manager application, you can access the Net-Net Route Management Central configuration setup option. If you are not licensed for Route Manager, this setup option will not be visible.

1. Enter 5 and press Enter. You are prompted about continuing.
2. Enter Y and press Enter to continue. You are prompted for the maximum number of route set backups.

Do you want to continue Yes/No?

=====

Route Manager Central configuration
Configure Route Manager Central properties

Configure number of route set backups per route set/backup type combination
Route Manager Central

Please enter the maximum number of route set backups per route set/backup type combination
of backups (1-500) [10]

3. Enter a value within range or press Enter to accept the default value. The Custom setup list of options appears. You can now configure SAML single sign-on.

Configuring SAML Single Sign-On

If you are licensed for the Route Manager application, you can access the SAML single signon configuration setup option. If you are not licensed for Route Manager, this setup option will not be visible.

The Route Manager application supports login through an external server using SAML single sign-on. You enter a username and password used in the request to the external server for authentication. If using self-signed certificates, you can import them into the Route Manager certificates file (cacerts).

To configure SAML single sign-on:

1. Press Enter to accept the default value 6. You are prompted about continuing.
2. Enter Y and press Enter. You are prompted for the username,

SAML Single sign on configuration
Provides SAML Single sign on authentication.

SAML Single sign on
Configure SAML Single sign on

Please enter the username for basic authentication to SAML Responder
Username [] █

3. Enter the username for basic authentication and press Enter. You are prompted for the password.

Please enter the password for basic authentication to SAML Responder
Password [] █

4. Enter the password required for basic authentication and press Enter. You are prompted about the connection timeout.

Please enter the connection timeout to the SAML Responder
Connection timeout (seconds) (5-60) [5] █

5. Retain the default value 5. You are prompted about importing a certificate.

Would you like to import a certificate?
[] 1 - Yes
[X] 2 - No [Default]

Please select an option [2] █

If you do want to import a certificate, see the following section for details.

6. Press Enter to retain the default value No. You are done configuring SAML single sign-on.
7. Press Enter to quit setup. You are prompted about continuing.
8. Enter Y and press Enter to quit the setup program.

Importing Certificates**To import certificates:**

1. Enter 1 and press Enter to import certificates. You are prompted about continuing.

Please select an option [2] 1

[X] 1 - Yes
[] 2 - No [Default]

Do you want to continue Yes/No? █

2. Enter Y and press Enter. You are prompted for the import method.

```
Choose the import method
[X] 1 - File [Default]
[ ] 2 - HTTP mode
```

```
Please select an option [1] █
```

3. Press Enter to accept the default File. (If you choose the HTTP mode instead, you need to enter the same information.) You are prompted about continuing.
4. Enter Y and press Enter to continue. You are prompted for the alias name of the imported certificate.

```
Do you want to continue Yes/No?Y
```

```
Please enter the alias name for the imported certificate
Alias name [] █
```

5. Enter the alias name and press Enter. For example, acmep.csr. You are prompted for the certificate file.

```
Please enter the alias name for the imported certificate
Alias name [] acmep.csr
```

```
Please enter the certificate file
File [] █
```

6. Enter the name of the certificate file and press Enter. For example, apkt.cer. The loading process occurs and the certificate is added to the keystore using the alias. You are prompted about adding another.
7. Press Enter to accept the default value No. (Or Enter 1 to continue and repeat the steps for adding more.) You are prompted about continuing.
8. Enter Y to continue and press Enter.

Mail Server

The Mail server setup option is visible even if you are not licensed for Element Manager. If you are licensed for Element Manager, you can setup the mail server credentials to enable the sending of emails to a targeted Microsoft Exchange and Gmail server.

If you are not licensed for Element Manager, you can bypass this setup option.

To configure the mail server:

1. Enter 7 and press Enter. You are prompted to continue.

2. Enter Y and press Enter. The mail server configuration options appear.

```

Do you want to continue Yes/No?Y

=====

Mail Server configuration
This option is used to configure mail server

Configure mail server
Mail server configuration can be used for trap notification.

Please choose to quit mail server configuration or apply new configuration
[X] 1 - Configure mail server      [Default]
[ ] 2 - Quit out of mail server configuration

Please select an option [1] ☐

```

3. Press Enter to configure the mail server. You are prompted to continue.
4. Enter Y and press Enter to continue. The status of the mail server indicates it is not configured and the mail server host options appear.

```

Do you want to continue Yes/No?
Mail server: Not configured

Use the following options to configure mail server host.
[X] 1 - Configure mail server host
[ ] 2 - Apply new mail server configuration [Default]
[ ] 3 - Cancel out and do not apply changes

```

Please select an option [1]

5. Press Enter to configure the mail server host. You are prompted to continue.
6. Enter Y to continue. You are prompted for the server DNS name.

```

Do you want to continue Yes/No?Y
Mail server: Not configured

Provide the DNS name.
Host name [ ] ☐

```

7. Enter the DNS name and press Enter. For example, mail.acmepacket.com. If configuring a Gmail server you might enter smtp.gmail.com. The mail server name is changed and the mail server configuration options appear.

```
Do you want to continue Yes/No?Y
Mail server: Not configured
```

```
Provide the DNS name.
Host name [ ] mail.acmepacket.com
Mail server host name has been changed
Mail from: Not configured
```

```
Use the following options to configure mail from.
[X] 1 - Configure mail from
[ ] 2 - Apply new mail server configuration
[ ] 3 - Cancel out and do not apply changes [Default]
```

```
Please select an option [1] █
```

8. Press Enter to configure the From address you want to use. You are prompted about continuing.
9. Enter Y and press Enter to continue. You are prompted for the mail from address.
10. Enter the address you want used for the From address and press Enter. For example, if sending to Microsoft Exchange account mailadmin@acmepacket.com. If sending to a Gmail account, mailadmin@gmail.com. You are prompted about the mail logon.

```
Do you want to continue Yes/No?Y
Mail from: Not configured
```

```
Provide the mail from.
Mail from [ ] mailadmin@acmepacket.com
Mail from has been changed
Mail logon required: Not configured
```

```
Use the following options to configure mail logon required or not.
[X] 1 - Configure mail logon required
[ ] 2 - Apply new mail server configuration
[ ] 3 - Cancel out and do not apply changes [Default]
```

```
Please select an option [1] █
```

11. Press Enter to configure a mail logon. You are prompted about continuing.
12. Enter Y and press Enter to continue. You are prompted whether to provide the mail logon required.

13. Press Enter if logon is not required. Or enter true and press Enter. If false, the mail server host configuration options appear. If true, the mail logon user configuration options appear.

```
Provide the mail logon required.
Mail logon required true/false [false] true
Mail logon required has been changed
Mail logon user: Not configured

Use the following options to configure mail logon user.
[X] 1 - Configure mail logon user
[ ] 2 - Apply new mail server configuration
[ ] 3 - Cancel out and do not apply changes [Default]

Please select an option [1] █
```

14. Press Enter to configure the mail logon user. You are prompted about continuing.
15. Enter Y and press Enter to continue. You are prompted for the mail logon user.
16. Enter the username and press Enter. For example, if sending to a Microsoft Exchange server mailrecipient@acmepacket.com. Or for Gmail, mailrecipient@gmail.com. The mail logon user status changes and the configuration options appear.

```
Do you want to continue Yes/No?Y
Mail logon user: Not configured

Provide the mail logon user.
Mail logon user [ ] mailrecipient@acmepacket.com
Mail logon user has been changed
Mail logon user password: Not configured

Use the following options to configure mail logon user password.
[X] 1 - Configure mail logon user password
[ ] 2 - Apply new mail server configuration
[ ] 3 - Cancel out and do not apply changes [Default]

Please select an option [1] █
```

17. Press Enter to configure the mail logon user password. You are prompted about continuing.
18. Enter Y to continue and press Enter. You are prompted for the password.

19. Enter the password the user enters when they logon and press Enter. (Nothing displays on the screen when you enter the password.) The logon user password status changes and the configure mail server host options appears.

```
Do you want to continue Yes/No?
Mail logon user password: Not configured
```

```
Provide the mail logon user password.
Mail logon user password [ ]
Mail logon user password has been changed
Mail server: abc
```

```
Use the following options to configure mail server host.
[ ] 1 - Configure mail server host
[X] 2 - Apply new mail server configuration [Default]
[ ] 3 - Cancel out and do not apply changes
```

```
Please select an option [2]
```

20. You can cancel the configuration without applying the changes or press Enter to apply the changes to the mail server configuration. You are prompted about continuing.
21. Enter Y to continue and press Enter. You are prompted to either apply the new configuration or quit.
22. Press Enter to quit the mail server configuration.

Starting the NNC Server

To start the server:

1. Change to the bin directory. For example:
`cd /opt/AcmePacket/NNC700/bin`
2. Execute the startnnc.sh script.
`./startnnc.sh`

Stopping the NNC Server

To stop the server:

1. Change to the bin directory. For example:
`cd /opt/AcmePacket/NNC700/bin`
2. Execute the shutdownnnc.sh script.
`./shutdownnnc.sh`

Migrating Data to Net-Net Central

You can migrate data to Net-Net Central from Net-Net EMS 6.x and Net-Net Route Manager Central 1.x.

Net-Net EMS 6.x

You can migrate the following information from 6.x to Net-Net Central 7.0

- User profile
- Alarms and events
- Device list and Net-Net SBC node information: IP addresses and passwords. Configuration data is not migrated.
- Banner text
- Audit trail logs

Net-Net Route Manager Central 1.x

You can migrate the same data listed for Net-Net EMS 6.x and existing data from the Net-Net Route Manager Central database.

Migration Process

Data migration from either Net-Net EMS 6.x or Net-Net Route Manager Central 1.x to Net-Net Central involves retrieving data from their databases and saving it to different categories of intermediate XML files. For example:

- alarms.xml
- UserSecurity.xml
- deviceDetails.xml

These XML files are compliant with the Berkeley DB XML (dbxml) files on the embedded Net-Net Central database. Those files are then exported to Net-Net Central where they are loaded into Berkeley dbxml files. When you run Net-Net Central, the migrated data is available.

Requirements

Review the following requirements before migrating data. If you plan to migrate data from both Net-Net EMS 6.x and Net-Net Route Manager Central 1.x, you must perform the migrations one at a time.

Net-Net EMS 6.X

The requirements for migrating data from Net-Net EMS 6.x include:

- Net-Net EMS 6.x database running
- Host name of the MySQL server
- MySQL port number
- Database user account with permissions allowing execution of SQL query
- User name and password of database user account
- Database name of Net-Net EMS data stored

Route Manager Central 1.x

The requirements for migrating data from Net-Net Route Manager Central 1.x include:

- Path to database entry folder. For example, on Linux:
- Stopping Route Manage Central

Before You Migrate Data

Before you migrate data, you need to run the setup.sh script to configure the Net-Net Central database and environment.

To run setup:

1. Login to the server.
2. Change directory to NNC700/bin. For example:

```
cd /opt/AcmePacket/NNC700/bin
```

3. Run the setup.sh script.

```
./setup.sh
```

You can apply the Net-Net Central license and complete the NNC installation in standalone or cluster mode.

If you try to migrate data without running setup first, the following message appears.

```
Starting Migration Application 2011-05-25 14:11:27,086
```

```
Please run NNC setup first. 2011-05-25 14:11:52,377
```

Data Migration from Net-Net EMS 6.x**To migrate data from Net-Net EMS 6.x:**

Ensure you have run setup.sh first and have shutdown Net-Net Central before you migrate data.

The default user groups Admin, Monitor, and Provisioner are not migrated. Net-Net Central creates those default groups with different names. Users are migrated but their passwords are reset to their usernames. When users login for the first time, they need to change their passwords.

1. Login as root.
2. Change directory to NNC700/bin. For example:

```
cd /opt/AcmePacket/NNC700/bin
```

3. Run the dataMigration.sh script to start the migration.

```
./dataMigration.sh
```

The migration options appear.

Migration Options:

```
1 EMS 6.x : Migrating data from EMS 6.x to NNC 7.x
```

```
2 RMC 1.x : Migrating data from RMC 1.x to NNC 7.x
```

```
3 Quit.
```

```
Select [1]
```

4. Press Enter to accept the default and migrate data from Net-Net EMS 6.x. You are prompted for the hostname or IP address.

Hostname or IP (required):

5. Enter the DNS name of the host or the IP address and press Enter. You are prompted for the port number.

Port [3306]:

6. Press Enter to accept the default value 3306 or enter a new port number and press Enter. You are prompted for the user name.

User [root]:

7. Press Enter to accept the default value or enter your username. You are prompted for the password.

Password:

8. Enter the password and press Enter. The data migration starts and the status appears on the screen.

Connected to database 2011-05-24 17:49:19,602

Migrating alarms data ... 2011-05-24 17:49:19,603

Migrating events data ... 2011-05-24 17:49:19,734

Migrating device groups data ... 2011-05-24 17:49:19,801

Migrating fault data ... 2011-05-24 17:49:19,907

Migrating ems details data ... 2011-05-24 17:49:19,934

Migrating device details data ... 2011-05-24 17:49:20,103

Migrating user security data ... 2011-05-24 17:49:20,142

Migrating audit trail logging data ... 2011-05-24 17:49:21,843

Start to load data into database 2011-05-24 17:49:22,104

Loading: alarms.xml into database ... 2011-05-24 17:49:22,530

Loaded: alarms.xml into database successfully. 2011-05-24 17:49:22,596

Loading: events.xml into database ... 2011-05-24 17:49:22,596

Loaded: events.xml into database successfully. 2011-05-24 17:49:22,640

Loading: AuditTrail.xml into database ... 2011-05-24 17:49:22,640

Loaded: AuditTrail.xml into database successfully. 2011-05-24 17:49:22,695

Loading: EMSDetails.xml into database ... 2011-05-24 17:49:22,949

Loaded: EMSDetails.xml into database successfully. 2011-05-24 17:49:23,031

Loading: Fault.xml into database ... 2011-05-24 17:49:23,032

Loaded: Fault.xml into database successfully. 2011-05-24 17:49:23,053

Loading: DeviceGroup.xml into database ... 2011-05-24 17:49:23,054

Loaded: DeviceGroup.xml into database successfully. 2011-05-24 17:49:23,093

Loading: DeviceDetails.xml into database ... 2011-05-24 17:49:23,093

Loaded: DeviceDetails.xml into database successfully. 2011-05-24 17:49:23,128

Loading: UserSecurity.xml into database ... 2011-05-24 17:49:23,128

Loaded: UserSecurity.xml into database successfully. 2011-05-24 17:49:23,227

Loaded data into database successfully 2011-05-24 17:49:23,227

Disconnected from database 2011-05-24 17:49:23,229

Completed data migration successfully! 2011-05-24 17:49:23,230

[nncentral@cosmo bin]#

Any new rules and attributes required by Net-Net Central are added during the migration process.

Migrating from Net-Net Route Manager Central 1.x

To migrate data from Net-Net Route Manager Central:

Ensure you have run `setup.sh` first and have shutdown Net-Net Central before you migrate data. The Route Manager Central data files need to be located on the same system as Net-Net Central.

1. Login as root.
2. Change directory to `NNC700/bin`. For example:
`cd /opt/AcmePacket/NNC700/bin`
3. Run the `dataMigration.sh` script to start the migration.

`./dataMigration.sh`

The migration options appear.

Migration Options:

```

1 EMS 6.x   : Migrating data from EMS 6.x to NNC 7.x
2 RMC 1.x   : Migrating data from RMC 1.x to NNC 7.x
3 Quit.
```

Select [1]

4. Enter 2 and press Enter to migrate data from RMC 1.x. You are prompted for the path to the `dbxml` files.

Path to RMC `dbxml` files (required):

5. Enter the path to the files and press Enter.

`/opt/ACMEPacket/rmc/db`

The database migration starts and progress information displays on the screen.

```

Migrating ems details data ... 2011-05-26 14:49:11,155
Migrating device details data ... 2011-05-26 14:49:11,467
Migrating user security data ... 2011-05-26 14:49:12,264
Migrating audit trail logging data ... 2011-05-26 14:49:12,295
Start to load data into database 2011-05-26 14:49:13,052
Loading: AuditTrail.xml into database ... 2011-05-26 14:49:13,177
Loaded: AuditTrail.xml into database successfully. 2011-05-26 14:49:16,863
Loading: EMSDetails.xml into database ... 2011-05-26 14:49:17,332
Loaded: EMSDetails.xml into database successfully. 2011-05-26 14:49:18,316
Loading: DeviceDetails.xml into database ... 2011-05-26 14:49:18,316
Loaded: DeviceDetails.xml into database successfully. 2011-05-26 14:49:18,621
Loading: UserSecurity.xml into database ... 2011-05-26 14:49:18,621
Loaded: UserSecurity.xml into database successfully. 2011-05-26 14:49:18,941
Loading: rms.xml into database ... 2011-05-26 14:49:18,949
Loaded: rms.xml into database successfully. 2011-05-26 14:49:19,245
Loaded data into database successfully 2011-05-26 14:49:19,245
Completed data migration successfully! 2011-05-26 14:49:19,245
```

Running Database Migration Again

You cannot run database migration a second time without re-initializing the Net-Net Central database first. Reinitializing the database permanently clears all the data it contains.

To reinitialize a database:

1. Login to the NNC server as root user.
2. Change directory to NNC700/bin. For example:
`cd /opt/AcmePacket/NNC700/bin`
3. Enter the following command to reinitialize the database:
`./reinitialize.sh`

Reinitialize Database with Data

```
./reinitialize.sh
Reinitialize options
[ ] 1 - Reintialize: This will clear all databases
[X] 2-Quit : Exit Reinitialize
Select an option [2] 1
Warning : This will delete all the content in database. Do you want to continue Yes/No
?yes
Deleting file ../../db/local/data/alarms.dbxml ...
Deleting file ../../db/local/data/AuditTrail.dbxml ...
Deleting file ../../db/local/data/__db.001 ...
Deleting file ../../db/local/data/__db.002 ...
Deleting file ../../db/local/data/__db.003 ...
Deleting file ../../db/local/data/__db.004 ...
Deleting file ../../db/local/data/__db.005 ...
Deleting file ../../db/local/data/__db.006 ...
Deleting file ../../db/local/data/events.dbxml ...
Deleting file ../../db/local/data/log.0000000001 ...
Deleting file ../../db/replicated/data/__db.001 ...
Deleting file ../../db/replicated/data/__db.002 ...
Deleting file ../../db/replicated/data/__db.003 ...
Deleting file ../../db/replicated/data/__db.004 ...
Deleting file ../../db/replicated/data/__db.005 ...
Deleting file ../../db/replicated/data/__db.006 ...
Deleting file ../../db/replicated/data/DeviceDetails.dbxml ...
Deleting file ../../db/replicated/data/DeviceGroup.dbxml ...
Deleting file ../../db/replicated/data/EMSDetails.dbxml ...
Deleting file ../../db/replicated/data/Fault.dbxml ...
Deleting file ../../db/replicated/data/log.0000000001 ...
Deleting file ../../db/replicated/data/UserSecurity.dbxml ...
Deleting file ../../db/local/migrationdb/alarms.xml ...
Deleting file ../../db/local/migrationdb/AuditTrail.xml ...
Deleting file ../../db/local/migrationdb/events.xml ...
Deleting file ../../db/replicated/migrationdb/DeviceDetails.xml ...
Deleting file ../../db/replicated/migrationdb/DeviceGroup.xml ...
Deleting file ../../db/replicated/migrationdb/EMSDetails.xml ...
```

Deleting file ../../db/replicated/migrationdb/Fault.xml ...
 Deleting file ../../db/replicated/migrationdb/UserSecurity.xml ...

Reinitialize Database without Data

```
./reinitialize.sh
Reinitialize options
[ ] 1 - Reintialize: This will clear all databases
[X] 2-Quit : Exit Reinitialize
Select an option [2] 1
Warning : This will delete all the content in database. Do you want to continue Yes/No
?yes
```

Data Migration Logging

Information about the data migration flow is captured in the Migration.log file located in the /AcmePacket/NNC700/logs directory.

The information includes:

- Net-Net EMS 6.x
 - MySQL database open/close connection
 - User management and security
 - Alerts
 - Events
 - Device Details
 - Device Groups
 - Audit trails
 - Fault
 - Banner
- Net-Net Route Manager Central 1.x
 - Berkeley database XML open/close containers
 - Database connection
 - User management and security
 - Device Details
 - Device Groups
 - Audit trails
 - Route sets
 - Route Management System (RMS)

Error Logging

Errors are logged as well as the data migration flow information. If errors in data migration occur, you need to reinitialize the database before you can retry the migration.

Deploying Migrated Data on Net-Net Central Cluster

To deploy data on the cluster members:

1. Run cold backup on the system on which the migration was successfully executed
2. Take the backup database files to the clustering system
3. Run restoration on the backup database files on any node that will be added to the cluster.

Mapping Device Groups to User Groups

After the data migration occurs, the devices exist in the NNC database but they might not be immediately visible in the Net-Net Central GUI. You need to map device groups to user groups for security purposes. Until that mapping is created, users in the user groups will not be able to see any of the devices in the device group.

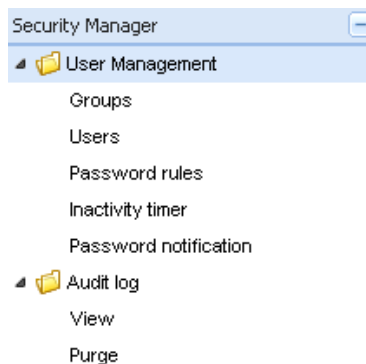
The Net-Net Central administrator must decide which of the device groups can be seen and managed by which of the default user groups that are created by default when Net-Net Central is installed:

- administrators: Super user group privileged to perform all operations
- LIAdministrators: Privileged to perform most operations including Lawful Intercept (LI) configuration changes
- provisioners: Privileged to configure Net-Net SBCs (if licensed for Element Manager) and save and apply the configuration with the exception of a LI configuration.
- monitors: Privileged to only view data, both configuration and other kinds of data.

Acme Packet recommends that all four of the default user groups have full visibility into all migrated device groups.

To map device groups to user groups:

1. Login to Net-Net Central as admin.
1. Expand the Security slider in the Navigation bar.
2. Expand the User Management directory.



3. Click Groups in the Navigation bar. A table of user groups appears in the content body. The table displays all of the operations categories with the current privilege levels for each user group.

- Click a user group name in the table and click Edit. For example, click LIAdministrators. The operations tabs appear.

Configuration		SBC system maintenance	Administrative operations	Fault management	Device group instances
Item					Privileges
▶ Configuration					Full

- Click Device group instances. The Device group instances tab lets you assign privileges and preview the device group hierarchy based on the privileges selected. That hierarchy will be reflected on the Device Manager slider.
- Click Set all to Full.

☐ Include children

The permissions on all device groups is set to Full. The device groups appear in the Preview column.

Device groups

Item	Privileges
Home	Full
▶ USA	Full
▶ EAST	Full
Boston	Full

Preview

Home

USA

EAST

Boston

- Click Apply.
- Repeat steps 4 through 7 to assign privileges to each user group.

Backup and Restore Database Servers

If you have administrator privileges, you can back up your servers either while they are shutdown or while still running. If you want to backup a server while it is running, tell any user working on that server to minimize their usage during backup.

Backing Up with Server Shutdown

Backing up while the server is shutdown is a cold backup. After all backups are complete, you can restart the servers.

1. Shutdown all servers you want to backup.
2. Log in as administrator.
3. Change directory to NNC700/bin. For example:

```
cd /opt/AcmePacket/NNC700/bin
```

4. Run the coldbackup script.

```
./backupcold.sh
```

The backup process runs.

Backing Up with Server Running

Backing up while the server is running is a hot backup. You need to tell all users of the server to minimize their system usage during hot backup process.

Note: Remember to tell any user on the server to minimize their system usage during backup.

1. Inform users to minimize their server usage.
2. Ensure the server is running.
3. Log in as administrator.
4. Change directory to NNC700/bin. For example:

```
cd /opt/AcmePacket/NNC700/bin
```

5. Start the hot backup script.

```
./backuphot.sh
```

The backup process runs.

Restoring Database Backups

You need to shutdown all servers on which you are restoring database backups. If restoring database backups on servers in a cluster, you need to restore databases on each one.

1. Shutdown all servers you plan to restore backups on.
2. Log in as administrator.
3. Change directory to NNC700/bin. For example:

```
cd /opt/AcmePacket/NNC700/bin
```

4. Start the database restore script.

```
./restoredb.sh
```

The database restoration process runs.

Configuring the Net-Net SBC for Net-Net Central Interaction

This section provides an example of how to configure the trap receiver and SNMP community in the Net-Net SBC to point to the NNC server. You need to configure these objects to enable Net-Net Central to provide fault management (SNMP traps), performance management statistics, and inventory control (SNMP) for this specific Net-Net SBC. If managing Net-Net 9000 SBCs, you need to create a virtual management interface to enable SOAP/XML (SNMP traps will also be sent to this virtual address).

You need to have Superuser privilege to configure your Net-Net SBC through a terminal by way of a local or remote connection.

Configuring the SNMP Interface

To configure the SNMP interface:

1. Connect to the Net-Net SBC and login.
2. Enable Superuser mode. For example:

```
User Access Verification
Password: <User Mode password>
ACMEPACKET> enable
Password: <Superuser Mode password>
ACMEPACKET#
```
3. Execute the **configure terminal** command.

```
ACMEPACKET# configure terminal
```
4. Execute the **system** command.

```
ACMEPACKET(configure)# system
```
5. Execute the **trap-receiver** command.

```
ACMEPACKET(system)# trap-receiver
```
6. Enter the following information:
 - ip-address <NNC server IP address>
 - filter-level <value>
 - community-name <value>

For example:

```
ACMEPACKET(trap-receiver)# ip-address 10.0.0.1
ACMEPACKET(trap-receiver)# filter-level all
ACMEPACKET(trap-receiver)# community-name acme
```
7. Create a trap receiver for each NNC server.
8. Enter **done**. For example:

```
ACMEPACKET(trap-receiver)# done
```
9. Enter **exit** to return to the system level.

```
ACMEPACKET(trap-receiver)# exit
```
10. Execute the **snmp-community** command.

```
ACMEPACKET(system)# snmp-community
```

11. Enter the following information:

- ip-address <NNC server IP address> or a list of IP addresses. For Net-Net 9000, ip-addresses add
- community-name <value>
- access-node <value> (leave at its default value)

For example:

```
ACMEPACKET(snmcommunity)# ip-address 10.0.0.1
ACMEPACKET(snmcommunity)# community-name acme
```

12. Enter **done**. For example:

```
ACMEPACKET(snmcommunity)# done
```

13. Enter **exit** to return to the system level.

```
ACMEPACKET(snmcommunity)# exit
```

14. Execute the **system-config** command.

```
ACMEPACKET(system)# system-config
```

15.

16. Enter the following information:

```
ACMEPACKET(system-config)# snmp-enabled enabled
```

17. Enter **done**. For example:

```
ACMEPACKET(system-config)# done
```

18. Verify the information.

19. Execute the **save-config** command (for Net-Net 9000, use the **save config** command).

```
ACMEPACKET(snmcommunity)# save-config
```

20. Execute the **activate-config** command to ensure SNMP is enabled (for Net-Net 9000, use the **activate** command).

```
ACMEPACKET# activate-config
```

21. Enter the **reboot** command for Net-Net 4000 SBC (no need to reboot Net-Net 9000).

```
ACMEPACKET(snmcommunity)# reboot
```

After the Net-Net SBC system restarts, execute the **show running-config** command. The output should contain a trap receiver and SNMP community objects with the information you just configured.

Starting the Net-Net Central Client and Connecting to the Server

Follow the instructions in this section to start the Net-Net Central client and log in to the server.

If logging into Net-Net Central when third-party X.509 certificates are used for HTTPS access, specify the hostname in DNS name format. Otherwise, the HTTPS you will have to click through security warnings about hostname mismatch between common name in the certificate and the IP address specified in the JNLP.

Note: You might experience difficulty connecting to Net-Net Central because of your network's proxy support. If you have trouble connecting, check the proxy settings for both your browser and the Java Webstart panel. See Verifying the Client System Settings for more information.

Verifying the Client System Settings

You should verify the client system has the required settings to connect to the NNC server. You need to have the Net-Net Central files from the distribution media at hand while verifying the client settings.

Verifying the Internet Explorer Browser Settings

If using Internet Explorer as your browser, you need to verify the following settings.

1. Open the Internet Explorer browser.
2. Choose the Tools menu and click Internet options.
3. Choose the Security tab.
4. Choose the Local intranet option and click Custom Level.
5. Enable the following options (if not already enabled) then click OK.
 - Run ActiveX controls and plug-ins under ActiveX controls and plug-ins
 - Active Scripting and Scripting of Java applets under Scripting
6. Choose the Internet option on the Security tab and click Custom Level. (This step is required if the client system accesses the NNC server via the Internet.)
7. Enable the following options (if not already enabled,) then click OK.
 - Run ActiveX controls and plug-ins under ActiveX controls and plug-ins
 - Active Scripting and Scripting of Java applets under Scripting
8. Click OK on the Internet options window to close it.

Disabling Proxy Server (Optional)

Follow these steps if your client system is configured as a proxy server and you do not want to use it for connecting with the Net-Net Central server.

1. Open the Internet Explorer browser.
2. Choose the Tools menu and click **Internet options**.
3. Click the Connections tab on the Internet options screen.
4. Click **LAN Settings** and then click **Advanced**.
5. Enter the Net-Net Central server IP address in the Exceptions panel.
6. Click **OK**.
7. Click **OK** on the Internet options window to close it.

Starting the Net-Net Central Client

You can start the Net-Net Central client by using either the HTTP or HTTPS login:

http://<NNC server IP address>:8080

https://<NNC server IP address>:8443 (self-signed certificates)

https://<domain name>:8443 (third-party X.509 certificates)

If using third-party X.509 certificates, use the DNS name of the host such as `nncserver.acmepacket.com` instead of the IP address. Then it matches the common name in the certificate.

Net-Net SBC License Information

The components of the Net-Net system software are licensed by Acme Packet, Inc. for your use. In order to use these components and deploy their related services in your network, you must have a valid license for each of them.

Licenses can be activated and deactivated in real time, and are fully extensible and upgradable. They are tied to specific Net-Net systems (by serial number) and cannot be transferred from one Net-Net system to another. Multiple licenses can be active on the same Net-Net system simultaneously. If the same feature happens to be covered by more than one license, then the latest expiration date applies.

Acme Packet software licenses are aggregate. This means that once a new license is added to the original license set, the related capacity, protocol, or interface becomes part of the functionality you can configure and deploy. For example, if your original license for session capacity is 1000 and then you add a new license for 3000 sessions, your new total session capacity is SBC.

The following software components, interfaces, and features are licensed. If you do not have a license for a given component, interfaces, or feature, its configuration parameters are not visible.

License	Description
Accounting	Establishes RADIUS servers to which the Net-Net 4000 SBC can make connections and send CDRs.
ACP	Enables the Net-Net 4000 SBC to respond to ACP requests. Required for Net-Net EMS use.
Administration Security	Enables the use of Administration Security features; installation and use of this feature set should be executed with care.
External Bandwidth Management	Enables interaction with external policy servers using COPS; you need this license if you want to use the resource allocation function (RACF)
External CLF Management	Enables interaction with external policy servers using COPS; you need this license if you want to use connectivity location function (CLF) support
External Policy Services	A combination of the External Bandwidth Management and External CFL Management licenses
H.248	Enables the H.248 ALG.
H.323	Enables H.323 signaling.
HA	Enables two Net-Net 4000 SBCs to work as an HA node so that, in case of failover, one system can take over for the other. The two systems paired as an HA node checkpoint configuration, signaling state, and media.
IDS	Enables the use of the Net-Net SBC's intrusion detection system (IDS).

License	Description
IKE	Enables the use of Internet Key Exchange version 1 (IKEv1).
IPv6	Enables IPv4-IPv6 interworking on your Net-Net 3800 or 4500; pure IPv6 works on these systems without the license being present.
IPSec	Enables the use of Internet Protocol Security (IPSec).
IWF	Enables SIP<—>H.323 IWF signaling. In order to run IWF between these two protocols, you must also have valid SIP and H.323 licenses.
LI	Enables lawful intercept use.
Load balancing	Establishes distribution of traffic across gateways, application servers, softswitches, etc.
MGCP	Enables MGCP/NCS signaling.
NSEP RPH	Enables support for Emergency Telecommunications Service (ETS), which gives priority treatment of National Security and Emergency Preparedness (NSEP) communications for IP network infrastructures.
QoS	Enables measurement for QoS (jitter, packet latency, and packet loss) on the Net-Net 4000 SBC.
Routing policies	Establishes routing policies on the Net-Net 4000 SBC. Release 4.1 introduces changes to the Acme Packet routing licence so that you can access more routing capability without obtaining a license. Without a routing license, you can view and set all local-policy-based parameters and specific parameters for only one policy attributes configuration (a subset of the local policy configuration). They are: • next-hop—Next signaling host IP address • realm—Realm of next signaling • action (formerly called replace-uri)—Replace Request-URI with next hop • app-protocol—Application protocol used to signal session agent. Without a routing license, the parameters noted in the list above appear in the local policy configuration instead of in the policy attributes configuration. You can also execute the ACLI test-policy command without a routing license. You need a routing license to specify multiple policy attributes per local policy, and to specify the remainder of the parameters in the policy attributes configuration, which are: • terminate-recursion—Whether or not to terminate route recursion with this next hop. • carrier—Carrier for the policy. • start-time—Daily time this policy goes into effect. • end-time—Daily time this policy is no longer in effect. • days-of-week—Days of the week this policy is in effect. • cost—(Unitless) cost for the policy. • state—State of the local policy attributes. • media-profiles—List of media profiles to use for this policy.
Session capacity	Determines the maximum number of sessions allowed by a Net-Net 4000 SBC for all protocols combined: SIP, MGCP, H.323, and SIP<—>H.323 IWF (interworking). Each flow that doubles back (or hairpins) through the Net-Net 4000 SBC counts as two flows. Options for session capacity are: 250, 500, 1000, 2000, 4000, 8000, 16000, and 32000. When your Net-Net 4000 SBC reaches 100% of its capacity, an alarm is generated and a trap sent.

License	Description
Session Replication for Recording (SRR)	Enables session replication for recording, which helps call centers record the signaling and media packets associated with their calls.
SIP	Enables SIP signaling.

There are two ways to access license information for your Net-Net Central servers:

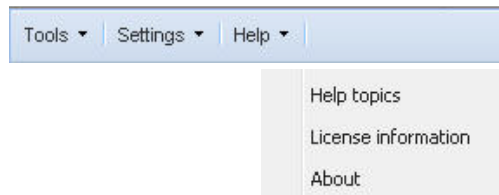
- Select License information from the Help menu in the top tool bar (shown below)
- Select License from the Device Manager navigation slider

Note: No privilege is required to view license information.

Accessing License Information

To access license information:

1. Click Help in the top tool bar. A drop-down menu appears.
2. Click **License information**.



The license information window appears.

License information

Select source:

License number 102356-1-43

Expiration Date

License Type permanent

Cluster Enabled

Applications

Application	Number of License Devices	In use
Element Manager	500	4
Route Manager	500	2

Signature

E/AAxhzeHUhdEZb1Yv45QzDOEES3niOIPd8+xpIyguDhEn4Fw
/3T8dxqkQffCaPI4ntovj/uDj6
fu8/eg80N+ZRbtWftO2pi5JCfWajMZxSorbhzujZ1KACD4H4iAvp89nRpniiIKMMLtECekp5V6R1
DON91tt28gBhTtHjojs0h4WYQhdXiOP2Dq8o1dT/D6HDLwnO+1dJlHqZAcR5jpSA5eMqk/GvIh
C8/mQ+yf33OG3Dol5v4yLaPkIiknrYIdu4zwMVQkeDEjwFWOCTuSCUC1KDz2EVJ0swXNzhkW86iQ
/NWpVkv9CVddSNAQkuVPVxGqHo3TbIrkmx

3. **Select source:**—Click the source IP address of the Net-Net Central server you want to view license information for in the drop-down list.

Select source:

The license information for this Net-Net Central server appears.

Note: For more information about the Net-Net Central license, please see the *Net-Net Central Core Functionality Guide*.

Unlicensed Net-Net Systems

If you log into a Net-Net system that is not licensed, you are warned that no licenses exist and that you need to enter a valid one. Until you enter a valid license, you can configure general system parameters, but not parameters for protocols and features.

When your Net-Net system arrives, you will need to obtain a key to activate the licenses for functionality you want to use. This original set of features is enabled with one key that you obtain from Acme Packet customer support at support@acmepacket.com.

Obtaining a License

If you choose to add functionality to your Net-Net system, each new feature will require its own key. To obtain additional licenses for functions on your Net-Net SBC, contact your customer support or sales representative directly or at support@acmepacket.com. You can request and purchase a license for the software you want, obtain a key for it, and then activate it on your Net-Net SBC.

When you obtain licenses, you need to provide Acme Packet with the serial number of your Net-Net system. You can see the system's serial number by using the ACLI **show version boot** command.

Trial Licenses

We also offer trial license periods for software components, allowing you to test a feature before deploying it.

Trial licenses are available for the same components listed at the beginning of this licensing section, but they only last for preset periods. After trial licenses expire, their functionality stops and configuration selections are removed. At that time, you can either stop using that particular functionality or you can purchase a license for it.

To obtain trial licenses, contact your Acme Packet sales or customer support representative directly or at support@acmepacket.com.

Loading a Configuration

You load a device's configuration by retrieving the device's configuration from the Net-Net SBC and loading it into the Net-Net Central database. To do this, you target the specific device you want to load from the Configuration Manager device table. This process is called loading a local configuration copy. It is an on-demand process, whereby you do not have to replicate large device configurations and retains the Net-Net SBC as the master database.

You must first associate the devices you add in Device Manager before you can load your device configurations. For more information about adding devices, see *Net-Net Central Core Functionality Guide, Managing Devices*.

Associating Devices

When you associate a device you link it to your Element Manager license. Your license allows you to configure a set number of devices at one time.

For more information about licensing, see the *Net-Net Central Core Functionality Guide, Viewing Net-Net Central License Information*.

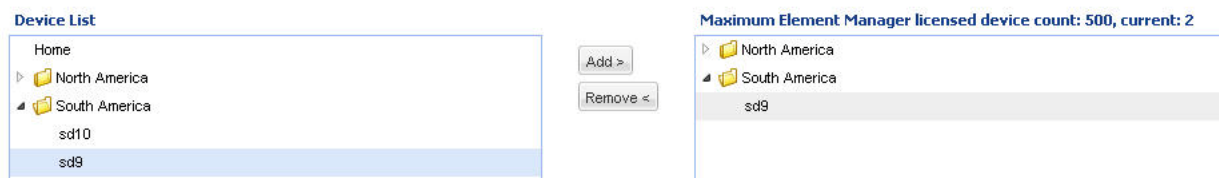
You can associate one device at-a-time, or you can associate all devices within a device group, if your license permits.

To associate a device to the Element Manager License:

1. Expand the Configuration Manager slider.
2. Click Devices. The Device table appears in the content area.
3. Click **Add devices**. The Devices associated with Element Manager license appears in the content area.
4. From the Device list, expand your device group folder and click the device you want to associate.
5. Click **Add >** to move your device to the Maximum Element Manager licensed device count table.

Devices associated with Element Management license

Select a device group or device from the Device list tree. Click Add to associate it with the Element Manager. Click Remove to cancel the association. You can only associate device groups and devices with the Element Manager if they have the required permissions.



6. Click **OK**. A success dialog box appears.
7. Click **OK**.

Your device is associated with your Element Manager license so you can load your device configuration.

Loading a Local Configuration Copy

To load a local configuration copy of your device for configuration:

1. Expand the Configuration Manager slider.
2. Click Devices. The Device table appears in the content area.
3. Click the arrow next to the device group folder to expand the list of devices within this device group.

Click the arrow to  East expand the view.

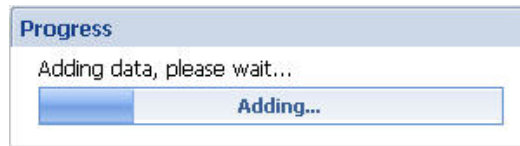
The list of devices appears for this device group.

Device	Target Name	Software Version	Hardware Version
USA			
West			
East			
 172.30.80.100	sd100	SC620m3	NN 4250
 172.30.80.210-1	sd210_sd211	SC620m3	NN 4250

4. Click the device you want to load and click **Load**.

East
 172.30.80.100 sd100

A Progress dialog box appears.



5. A Success dialog box appears. Click **OK**.



Your device's configuration is loaded and appears as a heading above the Devices table.

This is your loaded device. — **sd100(172.30.80.100)**

Device	Target Name	Software Version	Hardware Version	Device Co
USA				
West				
East				
172.30.80.100	sd100	SC620m3	NN 4250	211
172.30.80.210-1	sd210_sd211	SC620m3	NN 4250	9

After your Configuration is Loaded

Once your device's configuration copy is loaded, there are actions you can perform in Configuration Manger under Devices.

Performing Actions with Loaded Configurations

The actions are explained in the table below.

Action	Description
Refresh	Refreshes all devices for all device groups listed. The Refresh button is always enabled
Load	The Load button is enabled after you select a device from the list of devices on the Devices display Once invoked, loads the configuration for the currently-selected device for editing. If a new configuration version is available, the corresponding device's configuration data is refreshed
View Changes	Displays a list of all configuration changes for the selected device. By default, the table initially displays the changes made by the current user. By changing the User parameter to another user, or to the value, all, you can view all users' changes in a single list. From View Changes, with the appropriate permissions assigned, you can: - Refresh: Refreshes the data in the view changes list - Undo Changes: Will undo all changes you made to this device. You cannot undo changes made by another user - Change Owner: Allows you to transfer the ownership of your changes to another user. if you have admin privileges. You cannot change the ownership of changes made by another user - Update: Launches a dialog box with three options: Save and activate configuration (default), Save configuration, and Activate configuration. You cannot update the changes made by another user - Save to file: Saves the data displayed to a text file in comma separated values (CSV) format

Action	Description
Update	Launches a dialog box with three options: - Save and activate configuration (default): saves the configuration changes and activates the current configuration on the Net-Net SBC to make it the running configuration - Save configuration: Saves the current configuration changes to the Net-Net SBC (targeted device) Activate configuration: Activates the current configuration on the Net-Net SBC to make it the running configuration - Activate configuration: Activates the current save configuration on the Net-Net SBC to make it the running configuration The first two options are only available if there are pending changes to be saved. The third option is only available if there are no user changes, and there is a saved configuration pending activation.
View tasks	View the device tasks (or operations) performed on this device
Get Inventory	Accesses configuration inventory details for this device

About Devices Data

The data in the Devices table provides at-a-glance information pertaining to your loaded device in Configuration Manager.

sd10(172.43.254.110)

Device	Target Name	Software Version	Hardware Version	Device Configuration Version	Loaded Configuration Vers
Home					
172.43.254.110	sd10	SCX620m3p5	NN 4500	27	27
172.43.254.111	sd11	SCX620m3p5	NN 4500	23	23
172.43.254.112	sd12	SCX620m3p5	NN 4500	86	86

The table explains the data shown.

Data	Description
Device	The device (or HA pair) in the device group
Target Name	User-defined name for each device
Software Version	Full release version, including patch number, of software on device
Hardware Version	Full hardware platform identification
Device Configuration Version	The device's configuration version number. Note that until you load this device for the first time the device configuration number remains empty
Loaded Configuration Version	Configuration version number of the device's configuration stored locally in the database
Last Operation	Last operation performed on this device
Status	The status of the last operation performed on this device
Status Change Time	The local date and time of the last status update change on the device.

Removing an Associated Device

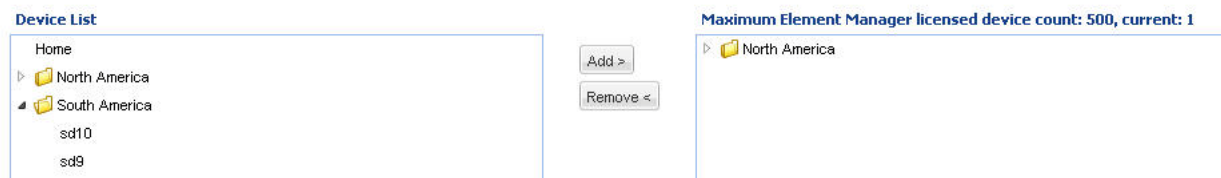
You can remove a device from the Maximum Element Manager licensed device count table, thereby removing the association to the Element Manager license.

To remove a device from the Element Manager License:

1. Expand the Configuration Manager slider.
2. Click Devices. The Device table appears in the content area.
3. Click **Add devices**. The Devices associated with Element Manager license appears in the content area.
4. From the Device list, expand your device group folder and click the device you want to remove.
5. Click **Remove <** to move your device from the Maximum Element Manager licensed device count table back to the Device List.

Devices associated with Element Management license

Select a device group or device from the Device list tree. Click Add to associate it with the Element Manager. Click Remove to cancel the association. You can only associate device groups and devices with the Element Manager if they have the required permissions.



Your device is removed from the Maximum Element Manager licensed device count table. If you remove the last device in a device group, the device group folder is also removed.

Relationship with ACLI

Net-Net Central provides a GUI-based approach to managing Net-Net SBCs. It provides the ability to configure and monitor standalone Net-Net SBCs and clusters of servers, with configuration, fault, performance, and security management for the Acme Packet product line.

The ACLI is an administrative interface that communicates with other components of the Net-Net system. The ACLI is a single DOS-like, line-by-line entry interface that you can use to configure and monitor your Net-Net family of products.

You can use Net-Net Central to perform almost all the same configuration and monitoring functions that can be performed using the ACLI (See the *Net-Net ACLI Reference Guide* for more information about using the ACLI). You can use both interfaces to work with Net-Net SBCs; even switching from Net-Net Central to login to a Net-Net SBC and use the ACLI.

Working with Configuration Views

There are three configuration views within Configuration Manager: Default, ACLI, and List. These views let you navigate around the top-level configuration elements in a specific way. The following sections describe each of these views.

No matter which view you select, you will be able to peruse all of the possible top-level configuration elements for your device. Once you select your view, and configuration element, the content area will display all attributes (or parameters) associated with this configuration element.

Navigating Between Configuration Views

You can switch between views at any time during your session. When you switch from one view to another view, the content area automatically opens to the top-level element you were working in from the previous view. For example, if you were working in the default view, under global settings, media manager, and you switched to the ACLI view, the content area will open to the top-level element, media-manager.

Below is an example of the Default view, with the top-level element, System, selected. The content area displays all attributes for the system element.

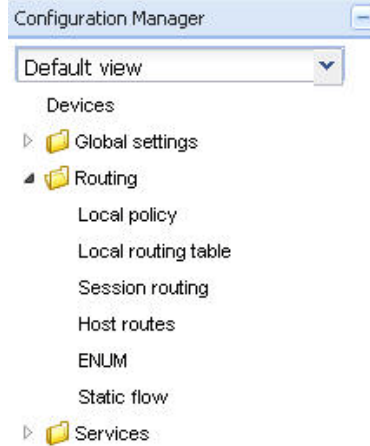
The screenshot displays the Session Element Manager Configuration Guide interface. The top navigation bar includes 'Tools', 'Settings', and 'Help'. The left sidebar shows a tree view with 'Global settings' expanded, and 'System' selected. The main content area is titled 'sd113(172.30.10.113)' and 'systemConfig'. It contains several configuration fields: 'Host name' (sd113), 'Description' (This is Net-Net SBC 113), 'Location' (QA lab upstairs), and 'Default gateway' (172.30.0.1) with a note '(Enter a valid IP address. Default:0.0.0.0)'. Below these is a 'Management interface access list' table with columns 'IP address', 'Netmask', and 'Description'. Under the table are 'Add', 'Edit', and 'Delete' buttons. Further down are several settings: 'Call trace' (disabled), 'Internal trace' (disabled), 'Log filter' (logs-fork), 'Restart' (enabled), 'reboot-exceptions' (empty), and 'Telnet timeout' (0). Each setting has a dropdown menu and a default value in parentheses.

Using the Default View

As the name implies, this is the first view presented, by default. With the default view, the top-level configuration elements are grouped logically, according to the type of configuration required: managing global settings and security, configuring routing, and defining services (media, signaling, and agents).

In the Default view example below all top-level configuration elements related to routing, such as local policy, session routing, or ENUM are grouped together under

Routing. You expand the configuration category folders (for example, Routing) to view the top-level elements for each category.



Understanding Default View Groups

The default view provides a level of organization not available with the other views. The logical grouping of top-level elements into function-specific categories allows you to approach your configuration tasks holistically, rather than just navigating to a top-level element individually. The following table displays the categories and their sub-categories. Top-level elements are found within these categories.

Configuration Categories	Sub-categories	Categories within Sub Categories
Global Settings	Management	
	IWF	
	Security	IKE
Routing		
Services	Media	
	Signaling	SIP, Translation, Call Admission Control
	Agents	

Using the ACLI View

With the ACLI view, the top-level elements are displayed as they would appear in the ACLI: grouped under media-manager, session-router, system, or security. All Net-Net Central configuration labels are listed according to their corresponding

ACLI parameter names (for example, **home-proxy-port**, and not **Proxy IP port**). The example below displays the top-level elements under media-manager.



Using the List View

With the List view, the top-level elements are displayed in an alphabetically-ordered list, and in the ACLI parameter format, for example, **enum-config** and not **ENUM**. There is no special grouping as with the other two views. This is an easy way to locate a top-level element if you are unsure of how it is grouped in the Default view or the ACLI view. For example:



User Interaction

This section contains information about user-interaction scenarios, error messages, tool tips, configuration for single-instance elements, and configuration timestamps.

Methods

Net-Net Central provides different methods for user interaction. Some examples of these methods are:

- Text fields
- Dialog boxes
- Combination boxes
- Checkboxes
- Options selector
- Radio buttons

Often instructions are provided for a parameter. These instructions help you to understand the guidelines of your entry, whether or not this is a required field, and provides the default value, if there is one. For example:

Max. message queue before failover (#) (Range:1..4096. Default:100)

Text Fields

Text fields allow you to manually enter information, usually with certain guidelines, for example, you can enter up to 25 characters, or you cannot include spaces in your entry. In the example below, there is a requirement to the right of the field to instruct you, which includes a default value for this parameter:

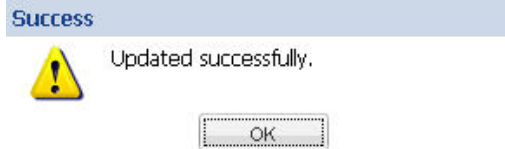
Default gateway (Enter a valid IP address. Default:0.0.0.0)

Dialog Boxes

There are three types of dialog boxes most commonly found in Net-Net Central:

- Alerts
- Informational
- Configuration-required

The simplest type of dialog box is an alert, which displays a message and can appear following an action, such as updating a parameter. This type of dialog box generally requires an acknowledgement, by clicking **OK**, to ensure the message has been read. For example,



Dialog boxes can deliver additional information. For example, if you view an event in the Event table under Fault Manager and want more information, click the event to open a dialog box with more event detail. For example,



Dialog boxes can also be a means to group parameters together, for example, all parameters required for adding a device. Typically, you open this type of dialog box by clicking a button, such as **Add** or **Edit**. You can encounter different user-input methods within a dialog box. For example:

Add Device [X]

Device type: 3800/4250/4500/9200 ▼

IP address 1: 172.30.10.100

IP address 2:

SNMP community name: public

SNMP port: 161

User name: admin

Password: ••••••••

Web protocol: HTTP ▼

Web port: 80

Web Services protocol: HTTP ▼

Web Services port: 80

Device Group: Home

Set device group

OK Apply. Add more? Cancel

Combination Boxes

With combination boxes, you expand a pre-defined list of choices by clicking on an arrow to the right of the text field. To make your selection, you click your choice in the list. For example:

*Protocol: UDP ▼ (Unique required. Default:UDP)

UDP

ICMP

ALL

TCP

Checkboxes

With checkboxes, you can select one or more choices from a pre-defined list of choices list by clicking in the box to check it, for example:

Password contains at least one of the following

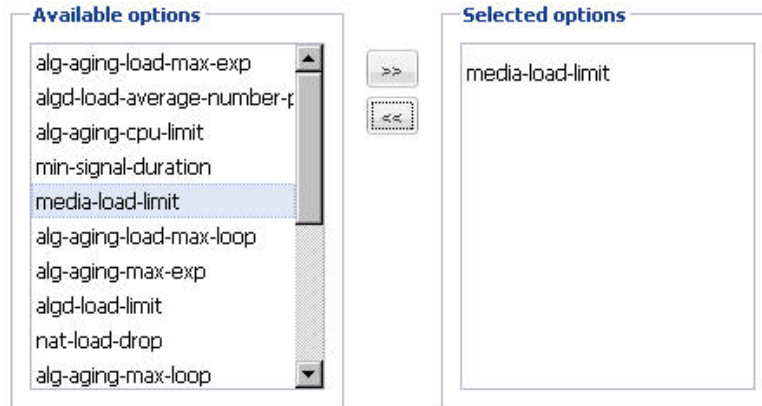
- ☒ Numeric character
- ☐ Alphabetic character
- ☒ Special character

Options Selector

When configuring options for certain Net-Net SBC features, you click the option in the Available options list to select it, and you click  to move it into the Selected options list. You repeat this process to add additional options.

To remove an option from the Selected options list, you click the option in the Selected options list to select it, and you click  to move the option back into the Available options list.

Below is an example of **media-load-limit** moved into the Selected options list:



User-Defined Options

Some options require additional configuration, for example, adding a percentage. In such cases, an options description provides you with guidelines for your entry. For example, the description for **media-load-limit** is:

Description

Syntax: media-load-limit=percent Defines the CPU% at which MBCD will start rejecting Flow ADD requests. The default is 95. A value of 0 or 100 disables load limiting.

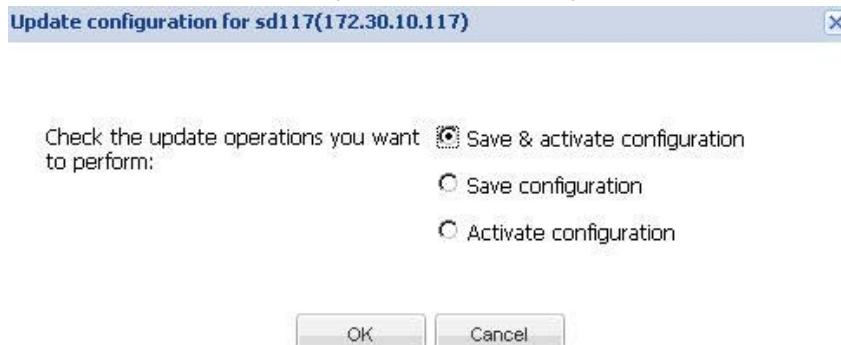
You enter text into the **User defined option** text field and click **Add** to update the option with your entry, for example:

User defined option

To edit a user-defined option, click **Edit** at the bottom of the Options selector dialog box.

Radio Buttons

With radio buttons, you select one choice only from a pre-defined list of choices. You click the radio button next to your choice to make your selection. For example:



Error Messages

If you incorrectly enter a value in a text field, and try to apply this change, you will receive an error message, along with an error panel detailing your input error. For example, if you enter a value beyond the valid value range, you will receive a validation error.

Entering an Incorrect Value

In the example below, an incorrect value was entered. Subsequently, the text field is outlined in red, with an exclamation point, letting you know there was an error:

Session cache timeout (hours) ! Range: 0..24. Default: 12)

An error message pops up in the content area directing you to an error panel with a description of your mistake, as shown in the example below. You must click **OK** to clear this error message.



The error panel is shown in the top-left corner of the content area. Below is the example of the error panel, Configuration Errors, associated with the **Session cache timeout (hours)** mistake referenced above:

Configuration Errors:

(1) Session cache timeout (hours)(sessionCacheTimeout): Invalid value '25'. : '25' is not a valid value of union type 'TLSGlobal_sessionCacheTimeout_uint'.

Reentering a Correct Value

Once you click **OK** to clear the error message, you can reenter a correct value in the text field. You must click **Apply** to apply your value. When correct, you will receive a success message. Click **OK** to clear the message.



Using Tool Tips

When configuring your device in Configuration Manager, position your cursor over a parameter field or checkbox to view a tool tip. Tool tips display the complete path to, and name of, the corresponding ACLI parameter. For example,

Prevent duplicate attributes (Default: disabled)

session-router->account-config->prevent-duplicate-attrs

Note: Tool tips will display no matter which view you use in Configuration Manager: Default, ACLI, or List view.

Configuring Single-Instance Configuration Elements

With multi-instance elements, the configuration parameters load automatically in the content area once you select your top-level element. When configuring single-instance configuration elements, such as accounting or session routing, you will be prompted to add the configuration to the content area by clicking **Add**. For example:

`sd100(172.30.80.100)`

Please press 'Add' button to add configuration

Once you click **Add** at the bottom of the content area, the accounting parameters load and a success message appears.

You must click **OK** to clear the message to begin configuring.

Modifying Primary Key Attributes

Net-Net Central validation rules will not allow you to modify a primary key attribute if the element-instance is referenced by another element instance. An asterisk (*) indicates a primary key attribute referenced by another element instance, and therefore, will not allow any modifications to the attribute. Primary key attributes can be modified if it has no references to another element instance.

Configuration Timestamp

You might see a configuration timestamp while configuring parameters. This timestamp is found in the content area, for example:

Last modified by NNC_admin_172.43.0.105

Last modified date 2011-02-17 16:25:03

The configuration timestamp displays last modified information for configuration changes made.

When you make configuration changes to elements, the Last modified by field displays the following information:

Last modified by:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine running Net-Net Central

Last modified date:

- Date and time, which is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS, for example 2011-02-17 16:25:03

Note: If a sub-element is modified, the timestamp belonging to the element is updated to indicate the modification time.

Saving and Activating Configurations

This section describes the following functions:

- Provisioning devices

- Viewing modifications
- Practicing suggested methods before saving and activating a configuration
- Saving and activating configurations to the device
- Locking and unlocking devices

Provisioning a Device in the LCC

Once you have loaded a device in the Configuration Manager, you can begin configuring your device. During the load process, the device's current saved configuration is uploaded to a local database. The local copy of the device's configuration is termed the Local Configuration Copy (LCC). LCC is an on-demand process that uploads configuration data once a device is targeted and loaded. NNC Element Manager uploads configurations to your LCC on-demand to ensure that the configuration devices managed by a NNC cluster are always synchronized without costly replication processes. This process also allows the user to work in an isolated configuration file that does not interrupt simultaneous configurations from other hosts. This makes the device the master database, and configurations are only saved and activated on the device at the request of the user.

It is important to remember that other administrative users can simultaneously provision configurations to the same device. Each user has their own view of their independent modifications known as the Local Configuration View (LCV). When a user renders a configuration page, the content displayed is a composite of the LCC and LCV. LCV allows several users to configure locally, and save and activate to the device at a later time. Upon saving a configuration to the device, the SBC validates the LCC and LCV locally. If the save is successful, the LCV is deleted. The next time the user targets the device, the updated LCC is uploaded from the device. The user must activate the configuration for this to become the running configuration for the SBC. Acme Packet recommends reviewing the Suggested Methods section below before saving and activating configurations to your device.

Below is the Devices Element screen of the Configuration Manager. You can see Devices selected in the Configuration Manager slider. The **View Changes** button at

the bottom of the content area loads the Local Configuration View (LCV) for viewing modifications made by the user.

The screenshot displays the Session Element Manager configuration interface. On the left is a navigation pane with a tree view under 'Default view' containing categories like Global settings, Management, Interfaces, Media manager, SIP, H248, H323, MF, Network parameters, Network management control, Security (with sub-items like TLS, IPSec, Public key, Lawful intercept, IMS AKA profile, IKE, Media security), Border gateway, Routing, and Services. The main content area is titled 'sd11_sd12(172.43.254.111-172.43.254.112)' and shows a table with columns: Device, Target Name, Software Version, Hardware Version, and Device Configuration Version. The table lists two regions: North America and South America. Under North America, a device is listed with IP 172.43.254.111, name sd11_sd12, software version SCX620m3p5, hardware version NN 4500, and configuration version 18. Under South America, no devices are listed. At the bottom of the interface are several buttons: Refresh, Load, View changes (highlighted in blue), Update, View tasks, Get inventory, and Add devices.

Device	Target Name	Software Version	Hardware Version	Device Configuration Version
North America				
172.43.254.111-172.43.254.11	sd11_sd12	SCX620m3p5	NN 4500	18
South America				

Viewing Modifications in the LCV

The LCV appears in the content area, and gives the user a copy of all modifications made to a device's configuration. The User combination box at the top of the content area allows you to view the LCV for other administrative users on the device.

Configuration changes

User:

admin

LCV

User	Type	Name	Operation	Time cha
admin	phy-interface	PHY01	created	2011-06-
admin	sip-config	sipConfig	created	2011-06-
admin	net-management-control	nm1	created	2011-06-

Once a user performs a successful save, the modifications will be removed from the LCV. At this time, configurations are pushed to the device, and a save task is present in the Device tasks table.

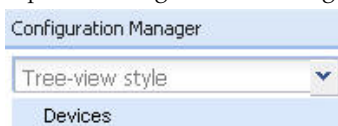
At the bottom of the LCV content area, there are five actions you can perform to the loaded device. The action buttons are explained in the table below.

Table 2 - 1:

Action	Description
Refresh	Refreshes all user changes for the device loaded. The Refresh button is always enabled.
Undo changes	Will undo all changes you made to this device. You cannot undo changes owned by another user.
Change owner	Allows you to transfer the ownership of one user's changes to another user if you have administrative privileges.
Update	Launches a dialog box with three options: - Save and activate configuration (default) - Save configuration - Activate configuration You cannot update the changes owned by another user. The first two options are only available if there are pending changes to be saved. The third option is only available if there are no user changes, and there is a saved configuration pending activation.
Save to file	Save to file: Saves the data displayed to a text file in comma separated values (CSV) format

To view modifications in the LCV:

1. Expand Configuration Manager and click Devices.



The Device table appears in the content area.

- Click the device group folder in which your device belongs to view the list of devices.
- Select the device.
- Click **View Changes**. The LCV screen appears in the content area.

Note: The device must already be loaded in Configuration Manager otherwise you will get an error message: Please load the device first.

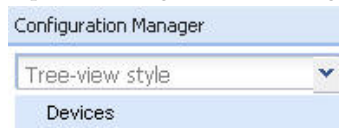
Viewing Task Logs

In the Devices Element screen, a user can click **View Tasks** to monitor the progress of save operations for the targeted device. The Device tasks table includes in-progress save tasks, and any tasks that are queued for saving. The progress of a task is tracked in the Status column.

A save task is given a task identifier that includes the user and time of operation.

To view tasks for the targeted device:

- Expand Configuration Manager and click Devices.



The Device table appears in the content area.

- Select the device whose save operations you'd like to view.
- Click **View Tasks**. The Device tasks table appears in the content area.

Note: The device must already be loaded in Configuration Manager otherwise you will get an error message: Please load the device first.

sd100(172.30.80.100)

Device tasks

Operation	Task identifier	Time stamp	Status	User name
SaveActivate	admin-2011-03-24-14-43	Thu Mar 24 10:43:56 EDT	inprogress	admin

At the bottom of the View Tasks content area are two actions you can perform. The action buttons are explained in the table below.

Table 2 - 2:

Action	Description
Refresh	Refreshes all tasks for the device loaded. The Refresh button is always enabled.
View Log	This button opens a detailed Save log for the selected task.

To view detailed save logs for the targeted device:

1. Select an Operation row from the Device tasks table and click **View log** for more information. The task log for this operation appears.

Note: If a row from the table is not selected and the button is clicked, an error dialog box appears.

Task log for sd100(172.30.80.100)

On host 172.30.80.12, SaveActivate task with id "admin-2011-03-24-14-43-55-394" is submitted @ Thu Mar 24 10:43:55 EDT 2011
Configuration version 32 from device 172.30.80.100 was loaded onto host 172.30.80.12 @ Thu Mar 24 10:43:55 EDT 2011

Performing Xsd model annotation validation on user changes in database. validation started @ Thu Mar 24 10:43:55 EDT 2011

Xsd model annotation validation completed. Status: SUCCESS @ Thu Mar 24 10:43:56 EDT 2011

publishing update task (SaveActivate) to processing queue. Publish to queue started @ Thu Mar 24 10:43:56 EDT 2011

On host 172.30.80.12, task thread is started to process "admin-2011-03-24-14-43-55-394" @ Thu Mar 24 10:43:56 EDT 2011

Configuration version 32 from device 172.30.80.100 was loaded onto host 172.30.80.12 @ Thu Mar 24 10:43:56 EDT 2011

Performing LOCK-DEVICE in SBC. lock device started @ Thu Mar 24 10:43:56 EDT 2011

Locked device completed. Status: SUCCESS @ Thu Mar 24 10:43:56 EDT 2011

Performing RESTORE-CONFIG. restore config started @ Thu Mar 24 10:43:56 EDT 2011
This may task few minutes. Please wait ...

RESTORE-CONFIG completed. Status: SUCCESS @ Thu Mar 24 10:43:57 EDT 2011

Update user changes to device started @ Thu Mar 24 10:43:58 EDT 2011

Element Type	Operation	Object Name	Status
enforcementProfile	ADD	Enforce1	SUCCESS
mediaPolicy	ADD	MedPo1	SUCCESS
systemConfig	MODIFY	systemConfig	SUCCESS

Updated total of 3 elements to device Completed. Status: SUCCESS @ Thu Mar 24 10:43:58 EDT 2011

Performing data integrity check on elements updated to device @ Thu Mar 24 10:43:58 EDT 2011

Data integrity check completed. Status: SUCCESS @ Thu Mar 24 10:44:30 EDT 2011

Performing SAVE-CONFIG. save config started @ Thu Mar 24 10:44:30 EDT 2011
This may task few minutes. Please wait ...

SAVE-CONFIG completed. Status: SUCCESS @ Thu Mar 24 10:44:31 EDT 2011

Performing ACTIVATE-CONFIG. activate config started @ Thu Mar 24 10:44:31 EDT 2011
This may task few minutes. Please wait ...

Refresh Save to file Close

Suggested Methods

When multiple administrative users have permission to save and activate configurations to a device, it is important to understand the processes of the tasks. Below are a few points about save and activate before getting started:

- Only one save and activate task can be completed at a given time for a targeted device. This is on a first in, first out basis. Multiple save and activate tasks can be performed simultaneously for separate devices.
- A user can see the progress of any save and activate operations in the Device tasks table.

- A user can view pending configuration modifications in the LCV.
- A user can view the history of the last 10 operations in the Device tasks table.

Understanding Configuration Models

The devices operate on dynamic data models that vary depending upon the software image of the device. The LCC loaded on your system reflects the configuration elements required for the model of your NNC device. It is possible that a software upgrade or downgrade might invalidate configuration modifications in your LCV, provided that they were created based on a different configuration model. Acme Packet recommends that you review the possible scenarios and best current practices provided below before performing the following actions:

- Saving and activating configurations to the device
- Performing a software upgrade or downgrade

Provisioning Cycle

Below is a summary of the tasks you might perform while provisioning your device:

1. Target and load the device in the Configuration Manager.
 - The device's current saved configuration is uploaded to the local database as the LCC.
2. Provision the device locally in the Configuration Manager.
 - The LCV displays the user's individual modifications to the LCC.
3. Lock the device before performing a save or activate task.
4. Save configuration.
 - During the save process, the SBC validates the LCC and LCV locally.
 - If validation succeeds, the save is successful. This configuration is now the saved configuration for the device, and will be used to generate an LCC for another user.
 - If validation fails, the save is unsuccessful and the user is notified.
 - If the save task succeeds, the user's configuration has been pushed to the device and the user's LCV is cleared.
5. Activate configuration.
 - The save configuration becomes the running configuration on the device.

Possible Scenarios for a Save to Device Failure

If a save to device operation fails, there are two scenarios that likely lead to a failure:

1. Invalid configuration due to one of the following reasons:
 - Configuration dependency check failure
 - Invalidated configuration data due to a software upgrade or downgrade

Note: A software upgrade or downgrade that invalidates a configuration will erase that user's LCV. It is important to read the Best Practices section before proceeding with a software upgrade or downgrade.

2. Network or device is unreachable
3. A device is locked

First Scenario

If the save to device operation fails, the Status column of the Device tasks table reads “Fail.”

- Review task logs to identify issues in the failed save operation
- Undo all changes in the LCV

In the first scenario, it is possible that the user’s LCC was synchronized with the device and the user’s modification conflicted with changes made by other NNC or ACLI users. This means that the user must target and load the device again, thereby synchronizing the LCC with the device.

In the event of a save to device failure due to a software upgrade or downgrade, the user will need to begin configurations again. This is due to the previous modifications having been created based on different configuration model. Synchronize your LCC by targeting and loading the device again.

Second scenario

In the second scenario, the user should retry the save and activate task. It is possible that a connection has been restored, and the operation will succeed.

In the event of failure due to an unreachable network or device, the user’s LCV is retained.

Third scenario

In the third scenario, check the Device Tasks table to see if an operation is currently in progress against the device. If a save operation’s status is Fail, and a lock is still in place, the administrative user can override the lock. Try the save operation again.

Best Current Practices

Before performing a software upgrade downgrade, check pending modifications in the LCV. You cannot do an update to the device for another user, and must notify the user if you would like them to run an update to the device before a software upgrade or downgrade. However, an administrative user can acquire the ownership of another user’s changes, and update those changes to the device before a software upgrade or downgrade.

An administrative user can lock a device before provisioning the LCC. A locked device disables save and activate updates from other users.

Applying Provisioning Changes in Net-Net Central

When working in Net-Net Central, you will routinely press the **Apply** following your configuration tasks. By pressing **Apply**, you are applying your changes to the Net-Net Central database. This does not apply these configuration changes to the device.

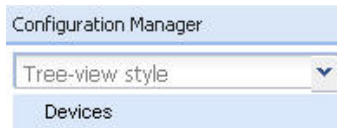
You must perform a **Save & Activate** to apply the changes to your device.

Saving and Activating Configuration

This section explains how to save the configuration changes to the Net-Net SBC and activate it. During the save process, other users are locked out of making changes to the targeted Net-Net SBC.

To save configuration changes to the Net-Net SBC:

1. Expand Configuration Manager and click Devices.



The Device table appears in the content area.

2. Click the device group folder in which your device belongs to view the list of devices.
3. Click the device whose configuration changes you want to save to select it.

Note: The device must already be loaded in Configuration Manager otherwise you will get an error message: Please load the device first.

4. Click **Update**. The Update configuration for SD dialog box appears.
5. **Check the update operations you want to perform:—Click one of the following:**
 - Save & activate configuration (default): Invokes the save/activate process
 - Save configuration: Invokes the save process
 - Activate configuration: Invokes the activate process

Note: The first two options are only available if there are pending changes to be saved.

The third option is only available if there are no user changes, and there is a saved configuration pending activation.

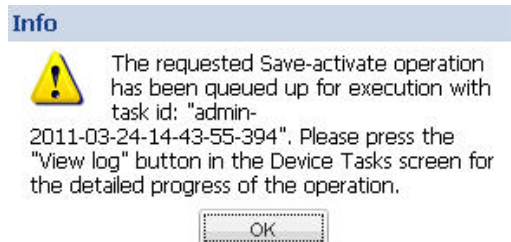


Check the update operations you want to perform:

☒ Save & activate configuration
☐ Save configuration
☐ Activate configuration



6. Click **OK**. An Info message appears, for example:



- Click **OK**. The operation you selected, for example, SaveActivate, appears in the Device tasks table.

sd100(172.30.80.100)

Device tasks

Operation	Task identifier	Time stamp	Status	User name
SaveActivate	admin-2011-03-24-14-43	Thu Mar 24 10:43:56 EDT	inprogress	admin

- Click the operation row and **View log** for more information. The task log for this operation appears.

Task log for sd100(172.30.80.100)

On host 172.30.80.12, SaveActivate task with id "admin-2011-03-24-14-43-55-394" is submitted @ Thu Mar 24 10:43:55 EDT 2011
 Configuration version 32 from device 172.30.80.100 was loaded onto host 172.30.80.12 @ Thu Mar 24 10:43:55 EDT 2011

Performing Xsd model annotation validation on user changes in database. validation started @ Thu Mar 24 10:43:55 EDT 2011

Xsd model annotation validation completed. Status: SUCCESS @ Thu Mar 24 10:43:56 EDT 2011

publishing update task (SaveActivate) to processing queue. Publish to queue started @ Thu Mar 24 10:43:56 EDT 2011

On host 172.30.80.12, task thread is started to process "admin-2011-03-24-14-43-55-394" @ Thu Mar 24 10:43:56 EDT 2011

Configuration version 32 from device 172.30.80.100 was loaded onto host 172.30.80.12 @ Thu Mar 24 10:43:56 EDT 2011

Performing LOCK-DEVICE in SBC. lock device started @ Thu Mar 24 10:43:56 EDT 2011

Locked device completed. Status: SUCCESS @ Thu Mar 24 10:43:56 EDT 2011

Performing RESTORE-CONFIG. restore config started @ Thu Mar 24 10:43:56 EDT 2011
 This may task few minutes. Please wait ...

RESTORE-CONFIG completed. Status: SUCCESS @ Thu Mar 24 10:43:57 EDT 2011

Update user changes to device started @ Thu Mar 24 10:43:58 EDT 2011

Element Type	Operation	Object Name	Status
enforcementProfile	ADD	Enforce1	SUCCESS
mediaPolicy	ADD	MedPo1	SUCCESS
systemConfig	MODIFY	systemConfig	SUCCESS

Updated total of 3 elements to device Completed. Status: SUCCESS @ Thu Mar 24 10:43:58 EDT 2011

Performing data integrity check on elements updated to device @ Thu Mar 24 10:43:58 EDT 2011

Data integrity check completed. Status: SUCCESS @ Thu Mar 24 10:44:30 EDT 2011

Performing SAVE-CONFIG. save config started @ Thu Mar 24 10:44:30 EDT 2011
 This may task few minutes. Please wait ...

SAVE-CONFIG completed. Status: SUCCESS @ Thu Mar 24 10:44:31 EDT 2011

Performing ACTIVATE-CONFIG. activate config started @ Thu Mar 24 10:44:31 EDT 2011
 This may task few minutes. Please wait ...

- Click one of the action buttons:

- **Refresh:** Refresh the data in the task log
- **Save to file:** Save the data to a local file
- **Close:** Close the task log

Locking Devices

With the appropriate Admin permissions, you can lock a device. When locked, other users cannot perform operations on this device. For example:

- Rebooting a device
- Updating a configuration
- Updating the device for route sets
- Pushing configuration changes to the Net-Net SBC

Note: Only users with override lock permissions granted can override your lock. Otherwise, the device must be unlocked by you.

To lock a device for configuration:

1. Expand the Device Manager slider and click Devices. The Device table appears in the content area.
2. Click the device you want to lock in the device content area.
3. Click **Lock**. A Confirm dialog box appears.



4. Click **Yes**.

The lock icon appears next to the locked device and **Lock** toggles to **Unlock**.

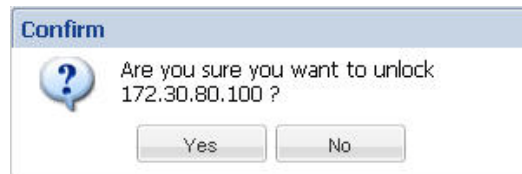
		172.30.80.100	sd100	SC620m3	NN 4250

Unlocking Devices

You can unlock a device if you are the owner of the lock, or if you have the appropriate admin permissions to override another user's lock on a device.

To unlock a device:

1. Click the locked device to select it.
2. Click **Unlock**. A Confirm dialog box appears.



3. Click **Yes**.
4. Click **Refresh**. The lock icon is removed. **Unlock** toggles to **Lock**.

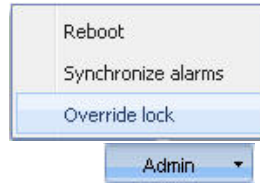
	172.30.80.100	sd100	SC620m3	NN

Overriding a Device Lock

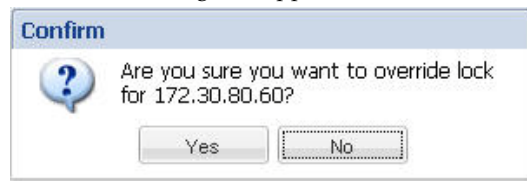
With the appropriate admin permissions assigned, you can override another user's lock on a device.

To override another user's lock on a device:

1. Expand the Device Manager slider and click Devices. The Device table appears in the content area.
2. Click the device whose lock you want to override in the device content area.
3. Click **Admin** to open a pop-up menu.
4. Click **Override lock**.



A Confirm dialog box appears.



5. Click **Yes**.
6. Click **Refresh**. The lock icon is removed from the device.

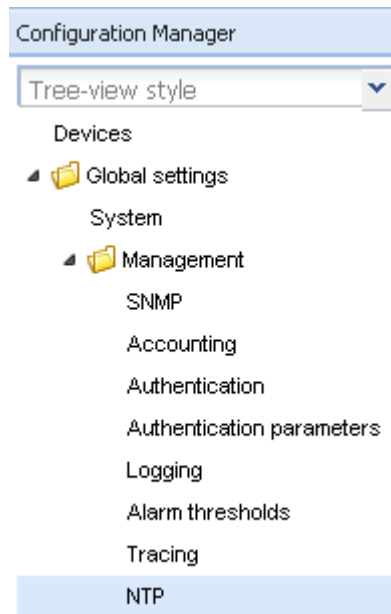
Setting Up NTP Synchronization

If you are using your Net-Net system for time-critical processing, you might want to use NTP for time synchronization. Setting NTP synchronizes both hardware and software clocks with the NTP server you specify.

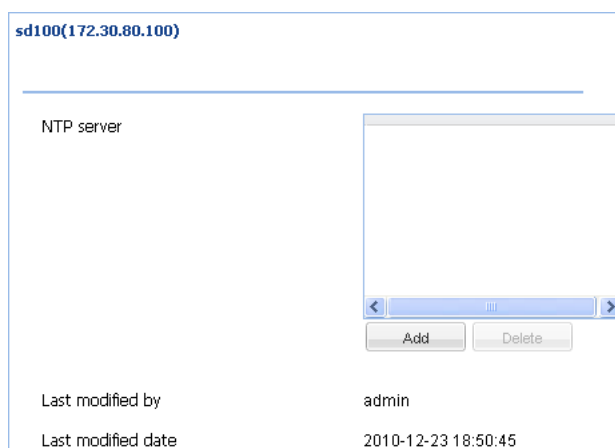
Note: Configuring NTP synchronization requires you reboot the Net-Net SBC.

To set NTP synchronization:

1. Expand the Configuration Manager slider.



2. Expand the Global settings folder.
3. Expand the Management folder
4. Click NTP. The NTP window opens in the content manager.



5. Click **Add**. The NTP server window appears.

6. **NTP server IP address**—Enter the IPv4 address of the server and click OK. The new server name appears in the IP address list.

Add ntp-sync->add-server

*NTP server address

(Unique required. Enter a valid IP address)

Apply

Cancel

7. Click Apply.
8. Reboot the Net-Net SBC.

RADIUS Authentication

A security feature that extends beyond the designation of ACLI User and Superuser privileges, the User Authentication and Access control feature supports authentication using your RADIUS server(s). In addition, you can set two levels of privilege, one for all privileges and more limited set that is read-only.

User authentication configuration also allows you to use local authentication, localizing security to the Net-Net SBC ACLI log-in modes. These modes are User and Superuser, each requiring a separate password.

The components involved in the RADIUS-based user authentication architecture are the Net-Net SBC and your RADIUS server(s). In these roles:

- The Net-Net SBC restricts access and requires authentication via the RADIUS server; the Net-Net SBC communicates with the RADIUS server using either port 1812 or 1645, but does not know if the RADIUS server listens on these ports
- Your RADIUS server provides an alternative method for defining Net-Net SBC users and authenticating them via RADIUS; the RADIUS server supports the VSA called ACME_USER_CLASS, which specifies what kind of user is requesting authentication and what privileges should be granted

How It Works

When RADIUS user authentication is enabled, the Net-Net SBC communicates with one or more configured RADIUS servers that validates the user and specifies privileges. On the Net-Net SBC, you configure:

- What type of authentication you want to use on the Net-Net SBC
- If you are using RADIUS authentication, you set the port from which you want the Net-Net SBC to send messages
- If you are using RADIUS authentication, you also set the protocol type you want the Net-Net SBC and RADIUS server to use for secure communication

Although most common set-ups use two RADIUS servers to support this feature, you are allowed to configure up to six. Among other settings for the server, there is a class parameter that specifies whether the Net-Net SBC should consider a specific server as primary or secondary. As implied by these designation, the primary servers are used first for authentication, and the secondary servers are used as backups. If you configure more than one primary and one secondary server, the Net-Net SBC will choose servers to which it sends traffic in a round-robin strategy. For example, if you specify three servers are primary, the Net-Net SBC will round-robin to select a server until it finds an appropriate one; it will do the same for secondary servers.

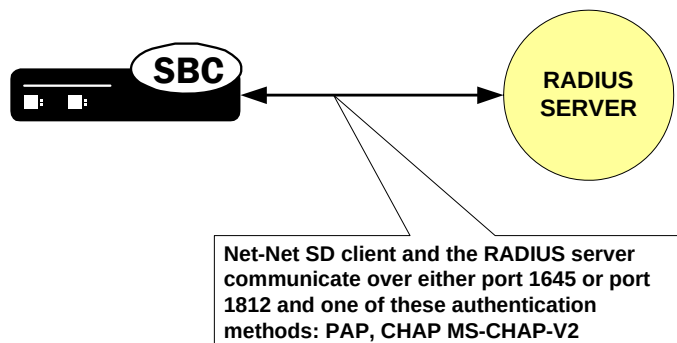
The VSA attribute assists with enforcement of access levels by containing one of the three following classes:

- None—All access denied
- User—Monitoring privileges are granted; your user prompt will resemble `ACMEPACKET>`
- Admin—All privileges are granted (monitoring, configuration, etc.); your user prompt will resemble `ACMEPACKET#`

Once it has selected a RADIUS server, the Net-Net SBC initiates communication and proceeds with the authentication process. The authentication process between the Net-Net SBC and the RADIUS server takes place uses one of three methods, all of which are defined by RFCs:

Protocol	RFC
PAP (Password Authentication Protocol)	B. Lloyd and W. Simpson, "PPP Authentication Protocols," RFC 1334, October 1992
CHAP (Challenge Handshake Authentication Protocol)	B. Lloyd and W. Simpson, "PPP Authentication Protocols," RFC 1334, October 1992 W. Simpson, "PPP Challenge Handshake Authentication Protocol (CHAP)," RFC 1994, August 1996
MS-CHAP-V2	G. Zorn, "Microsoft PPP CHAP Extensions, Version 2," RFC 2759, January 2000

Note: MS-CHAP-V2 support includes authentication only; password exchange is not supported or allowed on the Net-Net SBC.



PAP Handshake

For PAP, user credentials are sent to the RADIUS server include the user name and password attribute. The value of the “User-Password” attribute is calculated as specified in RFC 2865.

PAP Client Request Example

```
Radius Protocol
Code: Access Request (1)
Packet identifier: 0x4 (4)
Length: 61
Authenticator: 0x0000708D00002C5900002EB600003F37
Attribute value pairs
  t:User Name(1) l:11, value:"TESTUSER1"
    User-Name: TESTUSER1
  t:User Password (2) l:18, value:739B3A0F25094E4B3CDA18AB69EB9E4
  t:NAS IP Address(4) l:6, value:168.192.68.8
    Nas IP Address: 168.192.68.8(168.192.68.8)
  t:NAS Port(5) l:6, value:118751232
```

PAP RADIUS Response

```
Radius Protocol
Code: Access Accept (2)
Packet identifier: 0x4 (4)
Length: 20
Authenticator: 0x36BD589C1577FD11E8C3B5BB223748
```

CHAP Handshake

When the authentication mode is CHAP, the user credentials sent to the RADIUS server include “username,” “CHAP-Password,” and “CHAP-Challenge.” The “CHAP-Password” credential uses MD-5 one way. This is calculated over this series of the following values, in this order: challenge-id (which for the Net-Net SBC is always 0), followed by the user “password,” and then the challenge (as specified in RFC 1994, section 4.1).

CHAP Client Request Example

```
Radius Protocol
Code: Access Request (1)
Packet identifier: 0x5 (5)
Length: 80
Authenticator: 0x0000396C000079860000312A00006558
Attribute value pairs
  t:User Name(1) l:11, value:"TESTUSER1"
    User-Name: TESTUSER1
  t:CHAP Password (3) l:19, value:003D4B1645554E881231ED7A137DD54FBF
  t:CHAP Challenge (60) l:18, value: 000396C000079860000312A00006558
  t:NAS IP Address(4) l:6, value:168.192.68.8
    Nas IP Address: 168.192.68.8(168.192.68.8)
  t:NAS Port(5) l:6, value:118751232
```

CHAP RADIUS Response

```
Radius Protocol
Code: Access Accept (2)
Packet identifier: 0x4 (4)
Length: 20
Authenticator: 0x3BE89EED1B43D91D80EB2562E9D65392
```


MS-CHAP-v2 Handshake

When the authentication method is MS-CHAP-v2, the user credentials sent to the RADIUS server in the Access-Request packet are:

- “username”
- MS-CHAP2-Response—Specified in RFC 2548, Microsoft vendor-specific RADIUS attributes
- MS-CHAP2-Challenge—Serves as a challenge to the RADIUS server

If the RADIUS authentication is successful, the Access-Accept packet from the RADIUS server must include an MS-CHAP2-Success attribute calculated using the MS-CHAP-Challenge attribute included in the Access-Request. The calculation of MS-CHAP2-Success must be carried out as specified in RFC 2759. The Net-Net SBC verifies that the MS-CHAP2-Success attribute matches with the calculated value. If the values do not match, the authentication is treated as a failure.

MS-CHAP-v2 Client Request Example

Some values have been abbreviated.

```
Radius Protocol
Code: Access Request (1)
Packet identifier: 0x5 (5)
Length: 80
Authenticator: 0x0000024C000046B30000339F00000B78
Attribute value pairs
  t:User Name(1) l:11, value:"TESTUSER1"
    User-Name: TESTUSER1
  t:Vendor Specific(26) l:24, vendor:Microsoft(311)
    t:MS CHAP Challenge(11) l:18, value:0000024C000046B30000339F00000B78
  t:Vendor Specific(26) l:58, vendor:Microsoft(311)
    t:MS CHAP2 Response(25) l:52, value:00000000024C000046B30000339F00000B78...
  t:NAS IP Address(4) l:6, value:168.192.68.8
    Nas IP Address: 168.192.68.8(168.192.68.8)
  t:NAS Port(5) l:6, value:118751232
```

MS-CHAP-v2 RADIUS Response

```
Radius Protocol
Code: Access Accept (2)
Packet identifier: 0x6 (6)
Length: 179
Authenticator: 0xECB4E59515AD64A2D21FC6D5F14D0CC0
Attribute value pairs
  t:Vendor Specific(26) l:51, vendor:Microsoft(311)
    t:MS CHAP Success(11) l:45, value:003533s33d3845443532443135453846313...
  t:Vendor Specific(26) l:42, vendor:Microsoft(311)
    t:MS MPPE Recv Key(17) l:36, value:96C6325D22513CED178F770093F149CBBA...
  t:Vendor Specific(26) l:42, vendor:Microsoft(311)
    t:MS MPPE Send Key(16) l:36, value:9EC9316DBFA701FF0499D36A1032678143...
  t:Vendor Specific(26) l:12, vendor:Microsoft(311)
    t:MS MPPE Encryption Policy(7) l:6, value:00000001
  t:Vendor Specific(26) l:12, vendor:Microsoft(311)
    t:MS MPPE Encryption Type(8) l:6, value:00000006
```

Management Protocol Behavior

When you use local authentication, management protocols behave the same way that they do when you are not using RADIUS servers. When you are using RADIUS servers for authentication, management protocols behave as described in this section.

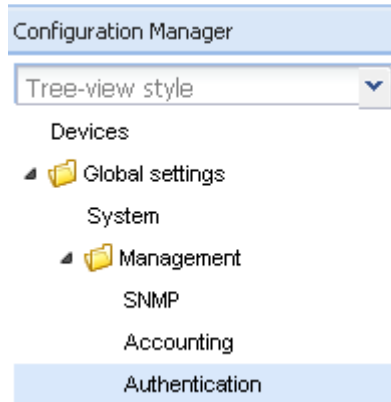
- FTP—If you set the user name to “user” or “admin,” the user is authenticated locally. Otherwise, configured RADIUS servers are used for authentication.
- SFTP in pass-through mode—If you do not configure an SSH account on the Net-Net SBC, the RADIUS server is contacted for authentication for any user that does not have the user name “user.” The Net-Net SBC uses local authentication if the user name is “user.”
- SFTP in non-pass-through mode—When you have configured an SSH account on the Net-Net SBC, user authentication takes place locally.

Configuring RADIUS Authentication

To configure RADIUS authentication support:

The parameters you set for individual RADIUS servers identify the RADIUS server, establish a password common to the Net-Net SBC and the server, and establish trying times. Setting the class and the authentication methods for the RADIUS servers can determine how and when they are used in the authentication process.

1. Expand the Configuration Manager slider.



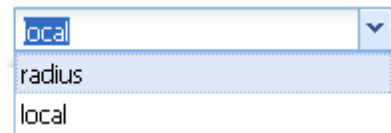
2. Expand the Global settings folder.
3. Expand the Management folder.
4. Click Authentication.
5. Click Add.
6. **Source port**—Choose the port number on the Net-Net SBC to send messages to the RADIUS server from the drop-down list.

Source port

1812

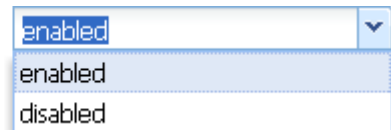
7. **Authentication model**—Choose the type of user authentication from the drop-down list.

Authentication model



8. **Allow local authorization**—Choose enabled from the drop-down list if you want the Net-Net SBC to authorize users to enter Super (administrative) mode locally even when your RADIUS server does not return the ACME_USER_CLASS VSA or the Cisco-AVPair VSA.

Allow local authorization



9. **Authentication protocol**—Choose the protocol type to use with your RADIUS server from the drop-down list.

Authentication protocol

A drop-down menu with a blue border. The selected option is 'pap'. The list of options includes 'pap', 'mschapv2', 'eap', 'chap', 'unknown', and 'pap' (highlighted at the bottom).

10. **Login as admin**—Choose enabled if you want users to be logged automatically in Superuser (administrative) mode.

Login as admin

A drop-down menu with a blue border. The selected option is 'enabled'. The list of options includes 'enabled' and 'disabled'.

RADIUS Server Settings

1. Scroll down the content manager to the RADIUS server section.
2. Click Add to configure a RADIUS server, or Edit to revise a current Radius server.
3. **Administrative state**—Select enabled to enable this RADIUS server.

Administrative state

A drop-down menu with a blue border. The selected option is 'enabled'. The list of options includes 'enabled' and 'disabled'.

4. **IP address**—Enter the IP address for the RADIUS server.

*IP Address

A text input field with a blue border containing the IP address '111.22.33.44'.

5. **Port**—Enter the port number on the remote IP address for the RADIUS server.

Port

A text input field with a blue border containing the port number '1812'.

6. **Shared secret**—Enter the password the RADIUS server and the Net-Net SBC share. This password is not transmitted between the two when the request for authentication is initiated.
7. **Network access server ID (NASID)**—Enter the network access server ID of the RADIUS server.
8. **Retry limit(seconds)**—Enter the number of times (not seconds) the Net-Net SBC retries to authenticate with this RADIUS server in seconds. Range of values is 1 to 5 times.
9. **Retry time(seconds)**—Enter the time in seconds the Net-Net SBC waits before retrying to authenticate with this RADIUS server. Range of values is 5 to 10 seconds.

10. **Maximum sessions to server**—Enter the maximum number of sessions to maintain with this RADIUS server. Range of values is 1 to 255 sessions.

Shared secret	acmepacket
Network access server ID(NASID)	89745
Retry limit(sec)	3
Retry time(sec)	5
Maximum sessions to server	255

11. **Class**—Choose the class of this RADIUS server as either primary or secondary. A connection to the primary server is tried before a connection to the secondary server is tried.
12. **Out of service period(seconds)**—Enter the time in seconds before the Net-Net SBC retries a RADIUS server that it has designated as dead. Range of values is 10 to 10000.
13. Click **Apply**.

Adding Authentication Methods

To add authentication methods:

1. In the Radius server table, click a RADIUS server row to select it.
2. Click Edit. The Edit Radius server dialog box appears.
3. **Supported authentication method**—Click Add. The Add Authentication methods dialog box appears.
4. **Authentication method**—Choose a authentication method the Net-Net SBC uses when communicating with the RADIUS server from the drop-down list.



5. Click Apply.

SSH RADIUS Authentication VSA Support

The Net-Net SBC supports the use of the Cisco Systems Inc.™ “Cisco-AVPair” vendor specific attribute (VSA). This attribute allows for successful administrator login to servers that do not support the Acme Packet authorization VSA. While using RADIUS-based authentication, the Net-Net SBC authorizes you to enter Superuser mode locally even when your RADIUS server does not return the ACME_USER_CLASS VSA or the Cisco-AVPair VSA.

For this VSA, the Vendor-ID is 1 and the Vendor-Type is 9. The list below shows the values this attribute can return, and the result of each:

- `shell:priv-lvl=15`—User automatically logged in as an administrator
- `shell:priv-lvl=1`—User logged in at the “user” level, and not allowed to become an administrator
- Any other value—User rejected

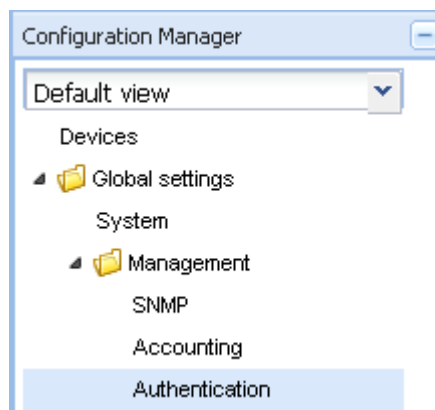
Configuring RADIUS Authentication Support

You enable this behavior in the authentication configuration by setting the **allow-local-authorization** parameter to **enabled**.

To configure RADIUS authentication support:

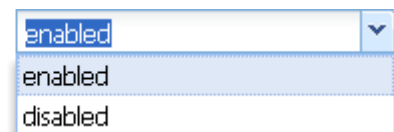
The parameters you set for individual RADIUS servers identify the RADIUS server, establish a password common to the Net-Net SBC and the server, and establish trying times. Setting the class and the authentication methods for the RADIUS servers can determine how and when they are used in the authentication process.

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Authentication.
3. Click Add.
4. **Allow local authorization**—Select enabled from the drop-down list if you want the Net-Net SBC to authorize users to enter Super (administrative) mode locally even when your RADIUS server does not return the ACME_USER_CLASS VSA or the Cisco-AVPair VSA.

Allow local authorization



5. Click Apply.

Introduction

This chapter explains how to configure system-level functionality for the Net-Net system. Both physical and network interfaces as well as general system parameters are required to configure your Net-Net SBC for service. Accounting functionality, SNMP configurations, trap configurations, and host routes are optional.

The following configurations are explained in this chapter:

- General system parameters—used for operating and identification purposes. In general, the informational fields have no specific effect on services, but are important to keep populated. The default gateway parameter is included here. It requires special attention since its configuration is dependent on the type of traffic the Net-Net SBC is servicing.
- Physical and network interfaces—enables the Net-Net SBC to communicate with any network element. Interfaces are one of the most basic configurations you need to create.
- SNMP—used for monitoring system health throughout a network.
- Syslogs and Process logs—used to save a list of system events to a remote server for analysis and auditing purposes.
- Host routes—used to instruct the Net-Net SBC host how to reach a given network that is not directly connected to a local network interface.

General System Information

This section explains the parameters that encompass the general system information on a Net-Net SBC.

System Identification

Global system identification is used primarily by the Net-Net SBC to identify itself to other systems and for general identification purposes.

Connection Timeouts

It is important to set administrative session timeouts on the Net-Net SBC for security purposes. If you leave an active configuration session unattended, reconfiguration access is left open to anyone. By setting a connection timeout, only a short amount of time needs to elapse before the password is required for Net-Net SBC access.

Timeouts determine the specified time period that must pass before an administrative connection is terminated. Any subsequent configuration activity can only be performed after logging in again to the Net-Net SBC. The timeout parameter can be individually specified for Telnet sessions and for console port sessions.

After the Telnet timeout passes, the Telnet session is disconnected. You must use your Telnet program to log in to the Net-Net SBC once again to perform any further configuration activity.

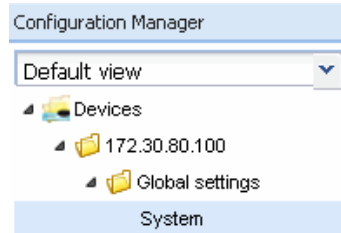
After the console timeout passes, the console session is disconnected. The current session ends and you are returned to the login prompt on the console connection into the Net-Net SBC.

Configuring General System Information

This section explains how to configure the general system parameters, timeouts, and the default gateway necessary to configure your Net-Net SBC.

To configure general system information:

1. Expand the Configuration Manager slider > Global settings.



2. Click System. The System window appears in the content area.
3. **Host name**—Enter the primary hostname used to identify the Net-Net system. This parameter is used by the software for informational purposes.
4. **Description**—Enter a textual description of the Net-Net system. This parameter is used for informational purposes.
5. **Location**—Enter a location description for your Net-Net system. This description is used for informational purposes. For example, you could include the site name and address of the location where the Net-Net system chassis is located.
6. **Default gateway**—Enter the default gateway for this Net-Net SBC. This is the egress gateway for traffic without an explicit destination. The application of your Net-Net SBC determines the configuration of this parameter.

Host name	AcmeSBC
Description	CUCM 7.0.2 <-> NNOS SCX6.2.0 MR-3 IOT
Location	Bedford, MA
Default gateway	172.43.254.1

7. Click **Apply**.
8. In the configuration slider, expand the Management folder.
9. Click the General tab.
10. **Telnet timeout**—Enter the time in seconds the Net-Net SBC waits when there is no Telnet activity before an administrative telnet session is terminated. A value of 0 disables this functionality.

11. **Console timeout**—Enter the time in seconds the Net-Net SBC waits when there is no activity on an administrative session before it terminates the session. A value of 0 disables this functionality.

Telnet timeout	900
Console timeout	900

12. Click **Apply** to continue.

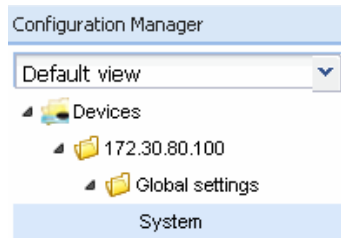
Configurable Timeout for Debug Logging

The configurable timeout for debug logging turns off the debug log level for any Net-Net SBC process enabled using the parameter, **Debug logging timeout (sec)**. This command does not effect log levels set in your configuration.

Configuring Timeout For Debug Logging

To configure a timeout for debug logging:

1. Expand the Configuration Manager slider > Global settings.



2. Click System. The System window appears in the content area.
3. **Debug timeout**—Enter the time in seconds you want to use for the debug timeout. This is the time allowed before the Net-Net SBC times out log levels for system processes set to debug.

The default and minimum value is 0, which disables this parameter. The maximum value you can enter is 65535.

Debug timeout	120

4. Click **Apply**.

Physical Interfaces

This section explains the different types of physical interfaces and how to configure them for use.

Overview

The Net-Net SBC's 10/100 and GigE interfaces provide physical connections to your network. Over the front interfaces, media and signaling traffic enter and exit the Net-Net SBC. The rear interfaces are used for management and high availability (HA).

You need to configure operating parameters for physical interfaces to support them on your Net-Net SBC. These values identify the name, location, basic Ethernet properties, and HA parameters; these are all configured on a per-interface basis.

Types of Physical Interfaces

There are two sets of physical interfaces on the Net-Net SBC.

- Front interfaces are on two removable physical interface cards
- Rear interfaces are directly connected to the Net-Net SBC chassis on three 10/100 Ethernet ports

Front Interfaces

The physical interface cards installed on your Net-Net SBC determine the number of interfaces, hardware protocol, and connection speed your Net-Net SBC can use for media and signaling traffic.

- The GigE physical interface cards offer either one or two ports per card, and can use single mode or multimode fiber with an LC connector.
 - Single-port GigE card—1 Gbps of bandwidth per configured port, totaling 2 Gbps total throughput with two cards installed on the chassis.
 - Two-port GigE card—1 Gbps of bandwidth per configured port, totaling 4 Gbps total throughput with two cards installed on the chassis.
- The 10/100 Ethernet physical interface card offers four ports per card, allowing eight 10/100 Ethernet connections.

For more information about physical interface cards, including installation and cabling, refer to the *Physical Interface Card Guide*.

Rear Interfaces

The first rear interface (wancom0) is used to carry traffic such as:

- SNMP
- Telnet
- SSH
- FTP
- ACP/XML
- Logs sent from the Net-Net SBC
- Boot the Net-Net SBC from a remote file server

The other two rear interfaces (port 1 and port 2) are used for state replication for HA. For HA, these rear interfaces on the Net-Net SBCs are directly connected by a crossover cable.

Note: To learn about HA, refer to this guide's "HA Nodes" chapter.

The following table summarizes the physical interface configuration parameters, which interface they are applicable to, and whether they are required.

Parameter	Front Interface	Rear Interface
name	R	R
operation type	R	R
port	R	R
slot	R	R
virtual mac	O	I
admin state	R	I
auto negotiation	R	I
duplex mode	R	I
speed	R	I
wancom health score	I	O

R = Required, O = Optional, I = Invalid

Before You Configure

Before you configure a physical interface:

1. Decide on the number and type of physical interfaces you need.
For example, you might have one media and signaling interface connecting to a private network and one connecting to the public network. You might also need to configure a maintenance interface for HA functionality.
2. Refer to HA nodes if you are configuring your Net-Net SBC for HA and follow the instructions there for setting special parameters in the physical interface configuration.

Configuring Physical Interfaces

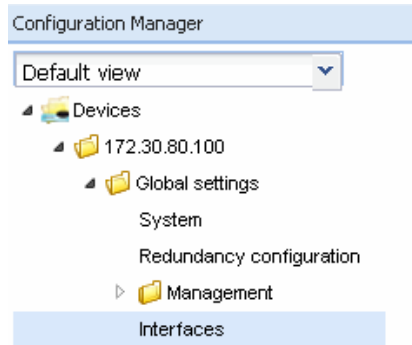
This section explains how to access and set the values you need to create a physical interface configuration. It also provides sample configurations for your reference.

Configuring Physical Interfaces

This section describes how to create physical interfaces on your Net-Net SBC.

To create and configure a physical interface:

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.
3. Click **Add**. The Add instance dialog box appears.
4. **Name**—Enter a name for the interface using any combination of characters entered without spaces. For example: **Internet** (for a Fast Ethernet media and signaling interface) or **maint0** (for a maintenance interface).
5. **Port**—Enter the port of this physical interface. From left to right as you face the chassis, the possible values are:
 - Front interfaces—**0 - 3**
 - Rear interfaces—**0 - 2**
6. **Operation type**—Select the type of physical interface from the combination box. For a front media and signaling interface, set this parameter to **media**. For a rear management interface, set this parameter to **maintenance** or **control**.
7. **Slot**—Enter the slot of this physical interface: **0** or **1**.
 - Front interfaces—0 is the left and 1 is the right

- Rear interfaces—0 is the only valid value

The 'Add instance' dialog box for 'PHYINTERFACECONFIG' contains the following fields:

Field	Value	Notes
*Name	s1p0	(Unique required)
*Port	0	(Range:0..3. Default:0)
*Operation type	Media	(Default:Control)
*Slot	1	(Range:0..1. Default:0)

Buttons: Apply, Cancel

8. Click **Apply**. A confirmation dialog box appears.
9. Click **OK** to dismiss the message and continue. The newly-added physical interface appears in the Physical interface table.

Configuring the Physical Interface

1. Select the new physical interface in the Physical interface table and click **Edit**. The Edit Physical Interface dialog box appears.

2. **Auto-negotiation**—Retain the default Auto-negotiation - 10/100Mbps value **enabled** so that the Net-Net SBC and the device to which it is linked can automatically negotiate the duplex mode and speed for the link.

If auto-negotiation is enabled, the Net-Net SBC begins to negotiate the link to the connected device at the duplex mode you configure. If auto-negotiation is disabled, then the Net-Net SBC will not engage in a negotiation of the link and will operate only at the duplex mode and speed you set.

3. **Duplex mode**—Choose the duplex mode from the combination box.

Given an operating speed of 100 Mbps, full duplex mode lets both devices on a link send and receive packets simultaneously using a total bandwidth of 200 Mbps. Given the same operating speed, half duplex mode limits the devices to one channel with a total bandwidth of 100 Mbps.

4. **Bandwidth (Mbps)**—Click a value in the combination box to set the bandwidth for this physical interface.

Auto-negotiation - 10/100Mbps

Auto-negotiation - 10/100Mbps	enabled	▼
Duplex mode	FULL	▼
Bandwidth (Mbps)	100	▼

You can also configure network interfaces in the Edit Physical interface dialog box. See *Configuring Network Interfaces* for detailed instructions.

5. Click **Apply**.

Interface Utilization: Graceful Call Control, Monitoring, and Fault Management

When you enable this feature, the Net-Net SBC monitors network utilization of its media interfaces and sends alarms when configured thresholds are exceeded. You can also enable overload protection on a per-media interface basis, where the Net-Net SBC will prevent call initializations during high traffic but still allow established calls to continue if traffic passes the critical threshold you define.

Calculation Overview

When enabled to do so, the Net-Net SBC performs a network utilization calculation for each of its media ports. This calculation takes into account rates of receiving and transmitting data, the speed at which each is taking place, and the quality of data traversing the interface. The Net-Net SBC keeps statistics for each media port so it can compare previously- and newly-retrieved data. For heightened accuracy, calculations are performed with milliseconds (rather than with seconds).

Alarms

In the physical interface configuration, you can establish up to three alarms per media interface—one each for minor, major, and critical alarm severities. These alarms do not have an impact on your system's health score. You set the threshold for an alarm as a percentage used for receiving and transmitting data.

For example, you might configure the following alarms:

- Minor, set to 50%
- Major, set to 70%
- Critical, Set to 90%

When the utilization percentage hits 50%, the system generates a minor alarm. At 70%, the system clears the minor alarm and issues a major one. And at 90%, the system clears the major alarm and issues a critical one. At that point, if you have overload protection enabled, the system will drop call initiations but allow in-progress calls to complete normally.

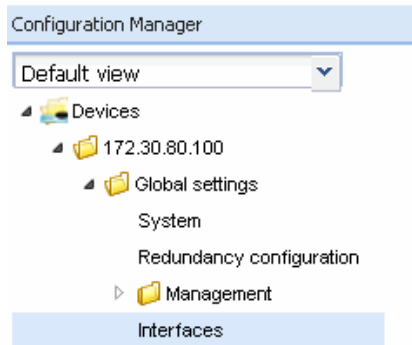
To prevent alarm thrashing, utilization must remain under the current alarm threshold for 10 seconds before the system clears the alarm and rechecks the state.

Configuring Utilization Thresholds for Media Interfaces

This section shows you how to configure alarm thresholds and overload protection per media interface.

To configure utilization thresholds for media interfaces:

1. Expand the Configuration Manager slider > Global settings.



2. Click **Interfaces**. The Network and Physical interfaces tables appear in the content area.
3. Select the interface you want to configure in the table and click **Edit**. The Edit Physical interface window appears.
4. Scroll to the Alarm threshold table.
5. Click **Add**. The Add instance dialog box appears.
6. **Alarm type**—
7. **Severity**—Select a severity value from the combination box for the alarm you want to define for this interface. The default for this parameter is minor.

8. Click **Apply**. The alarm threshold now appears in the Alarm threshold table.
9. Select the alarm in the table and click **Edit**. The Edit alarm threshold window appears.
10. **Value (%)**—Enter the percentage of utilization (transmitting and receiving) that you want to trigger the alarm. For example, you might define a minor alarm with a utilization percentage of 50.

Value (%)

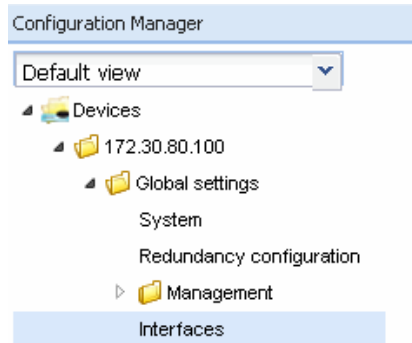
11. Click **Apply**.
12. Repeat steps 5 through 12 to add other threshold alarms.

Configuring Graceful Call Control

You can enable the Net-Net SBC to stop receiving session-initiating traffic on a media interface when the traffic for the interface exceeds the critical threshold you define for it.

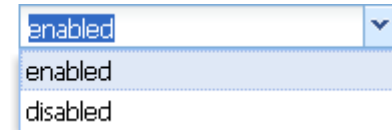
To enable graceful call control:

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.
3. Select the interface you want to configure in the table and click **Edit**. The Edit Physical interface window appears.
4. **Overload protection**—Select enabled from the combination box to enable graceful call control. Leave it set to disabled (default) if you do not want to use this feature.

Overload Protection



5. Click **Apply**.

SNMP Support

The apSysMgmtPhyUtilTable in the System Management MIB (apSysMgmtInterfaceObjects) supports this feature. Entries in the table are indexed by ifIndex for the specific port. See the *Net-Net 4000 MIB Reference Guide* for further information.

Trap Definitions

When any threshold is reached, the Net-Net SBC generates a trap containing this information:

- ifIndex—ifIndex of the port that reached the threshold
- CurrentValue—Value that is exceeding one of the thresholds
- MinorThreshold—Configured minor threshold; reports 0 if not configured
- MajorThreshold—Configured major threshold; reports 0 if not configured
- CriticalThreshold—Configured critical threshold; reports 0 if not configured
- RejectOverUtil—Boolean that reports 1 when session-initiating traffic is being blocked; reports 0 if disabled

See the *Net-Net 4000 MIB Reference Guide* for further information.

HDR Statistics

HDR supports a group name called **network-util** that reports interface utilization statistics. The Net-Net SBC only collects data for configured, valid ports.

Network Interfaces

This section describes the use and configuration of network interfaces.

Overview

The network interface element specifies a logical network interface. At least one network interface must be configured for each physical interface that already exists. When multiple network interfaces are configured on one physical interface, the configuration is known as a channelized physical interface, otherwise called a VLAN.

A default network interface element must be defined for every configured physical interface. The default network interface element is indicated by a subport ID field with a value of zero. Network Interfaces is a multiple instance configuration element. The combination of the name field and the subport ID field must be unique in order to identify a discrete network interface.

IP Configuration

A Net-Net SBC network interface has standard parameters common to nearly all IPv4 network interfaces. There are a few fields that are unique to the Net-Net SBC.

VLANs

VLANs are used to logically separate a single physical interface into multiple network interfaces. There are several applications for this like MPLS VPNs (RFC 2547), MPLS LSPs, L2VPNs (IPSec, L2TP, ATM PVCs), reusing address space, segmenting traffic, and maximizing the bandwidth into a switch or router. The range of services and management capabilities you can implement with VPNs is huge.

The primary applications of VLANs on the Net-Net SBC are VPNs and peering. Several peering partners may terminate their connections to a Net-Net SBC on a single physical interface. VLAN tags are used to segregate and correctly route the terminated traffic. The Net-Net SBC can support a maximum of 1024 VLANs per physical interface. Ingress packets that do not contain the correct VLAN tag will be dropped. All packets exiting on an egress interface will have the VLAN tag appended to them.

The Net-Net SBC can be included in an MPLS network through its connectivity to a PE router, which maps a MPLS VPN label to an 802.1q VLAN tag. Each Net-Net SBC with a Fast Ethernet or Gigabit Ethernet interface can terminate different 802.1q VLANs into separate network interfaces, each of which can represent a different customer VPN.

VLAN Network Layer Segmentation

VPNs are used to segment traffic at the network layer. Locally, a network is defined by the Net-Net SBC as a network interface or 802.1q VLAN. Each VLAN can be bridged into a Layer 2 VPN (Ethernet VLAN, Metro VPN, ATM VC, FR DLCI), a Layer 3 Routed VPN (i.e., MPLS VPN or LSP), or may simply be used to identify a traffic class (using VLANs to segregate traffic by customer or class of service). Separation of traffic implicitly provides a level of security.

Overlapping Networks

Overlapping networks are when two or more private networks with the same addressing schemes terminate on one physical interface. The problem this creates can easily be solved by using VLAN tagging. For example, two 10.x.x.x networks terminating on one Net-Net SBC network interface will obviously not work. The Net-Net SBC includes the IPv4 Address, IPv4 Subnet Mask and 802.1q VLAN tag in its Network Interface determination. This allows Net-Net SBC to directly interface to multiple VPNs with overlapping IPv4 Address space.

HIP

By default, the Net-Net SBC's FTP, ICMP, SNMP, and Telnet services cannot be accessed via the front network interfaces. In order to enable these services, the Net-Net SBC includes four fields that enable administrative traffic over the front interfaces. These are collectively known as the HIP, or host-in-path functions. The HIP parameters are effectively firewall functions that open the well-known ports for specified services on front interfaces.

Configuring Network Interfaces

This section explains how to access and configure network interface. It also provides sample configurations for your reference.

Special Considerations

Configuration changes to network interface parameters might have an impact on boot configuration parameters. After configuring the network interface, you might receive a message indicating that you could be changing boot config parameters under the following circumstances:

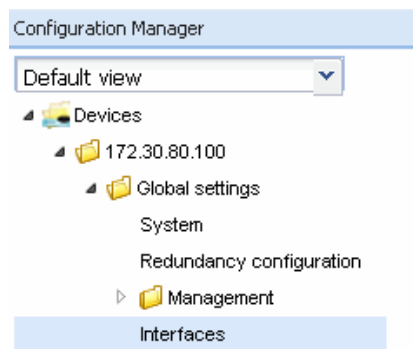
- A physical interface or network interface element matches the boot interface (for example, the physical port is the same as the boot port).
- The boot configuration parameters are modified, because the IPv4 address, netmask, or gateway is different from the corresponding boot configuration parameters.

You are asked if you want to continue. If you enter yes, the configuration will be saved and then the differing boot configuration parameters will be changed. If you enter no, then the configuration is not saved and the boot configuration parameters are not changed.

Configuring the first rear physical and network interface is optional because that interface, wancom0, is implicitly created by a valid bootparam configuration that specifies the boot device, IPv4 address, subnet, and gateway.

Creating a Network Interface**To create a network interface:**

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.
3. Select the physical interface to which the network interface will correspond and click **Edit**. The Edit Physical interface window appears.
4. Click **Add** in the Network Interface section. The Add instance dialog box appears.

5. **Physical interface**—Select the physical interface to which this network interface corresponds from the combination box.
6. **VLAN number**—Enter the subport ID as the VLAN number. If this network interface is not channelized, leave this set to 0. If this network interface is channelized, enter the appropriate VLAN tag.



The 'Add instance' dialog box for 'NETWORKINTERFACE' contains the following fields:

- *Physical interface:** A dropdown menu with 's1p0' selected. To the right, it says '(Unique required)'.
- *VLAN number:** A text input field with '0' entered. To the right, it says '(Unique required. Range:0..4095. Default:0)'.

At the bottom of the dialog are 'Apply' and 'Cancel' buttons.

7. Click **Apply**.

The new network interface appears in the Network Interface table.

Configuring the Network Interface

1. Select the network interface you want to configure and click **Edit**. The Edit Network interface window appears.
2. **IP address**—Enter the IP Address of this network interface.
3. **Subnet mask**—Enter the subnet mask of this network interface.

Host

Host name	phyTest
IP address	172.43.30.220
Subnet mask	255.255.255.0

4. **Primary IP address**—Enter the primary gateway that this network interface uses to communicate with the next hop.
5. **Secondary IP address**—If needed, enter the secondary gateway of this network interface.

Gateways

Primary IP Address	172.43.30.221
Secondary IP Address	0.0.0.0

6. **Primary**—Enter the DNS server of this network interface.
7. **First backup**—If needed, enter the secondary DNS server of this network interface.
8. **Second backup**—If needed, enter the third DNS server of this network interface.

9. **Default domain name**—Enter the default domain for use with DNS queries.

DNS

Primary	192.168.44.55
First backup	0.0.0.0
Second backup	0.0.0.0
Default domain name	acmepacket.com
DNS timeout	11

10. Scroll to the Host-In-Path Interfaces area.

Configuring a HIP Interface

To configure administrative service functionality on a front interface, you must define the IPv4 addresses on the front physical interfaces of your Net-Net SBC where you will receive administrative traffic. Adding HIP entries automatically opens the well-known port associated with a service.

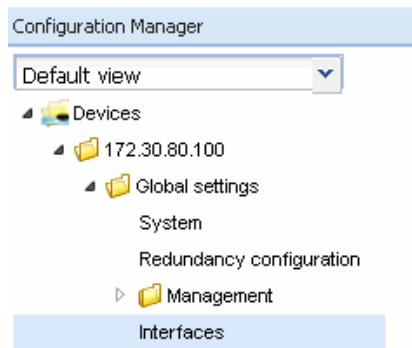
1. Click **Add** in the Host-In-Path Interfaces section of the Network interface window to create a new HIP interface. The Add instance window appears.
2. **IP Address**—Enter the IPv4 address on this network interface that is allowed to pass traffic to the host. Entries are IPv4 addresses of front panel network interfaces. Enter the appropriate IPV4 address that corresponds to the type of traffic.
 - **SNMP**: Set the IPv4 address where port 161 is opened. This lets SNMP traffic enter the Net-Net SBC and reach the host.
 - **Telnet**: Set the IPv4 address where port 23 is opened for Telnet access.
 - **FTP**: Set the IPv4 address where ports 20 and 21 are opened. This lets standard FTP packets enter the Net-Net SBC and reach the host.
 - **ICMP**: Set the IPv4 address to pass standard ping packets to the host.
3. Click **Apply** to accept these HIP values.

4. Click **Apply** at the bottom of the network interface window to complete network interface configuration.

Adding a Network Interface Description

To add a network interface description:

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.
3. Select the network interface interface from the table and click **Edit**.
4. **Description**—Enter a textual description for this network interface.

Description

network description

5. Click Apply.

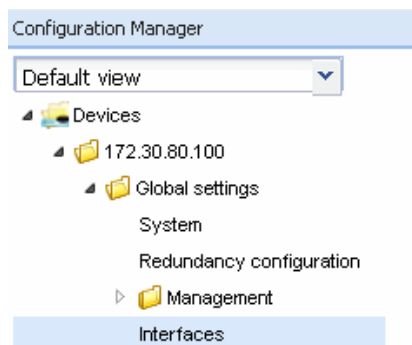
Configuring Multiple ICMP Addresses

You can configure multiple ICMP ping addresses on a single network interface, and you can designate a network interface and address from which to initiate pings.

When you configure multiple ICMP addresses in a network interface configuration, you must also configure the host-in-path addresses in the hip-ip-list for each ICMP address. For security, if the ICMP address and the hip-ip-list are not added for an address, the Net-Net 400 hardware discards ICMP requests or responses for the address.

To add an ICMP ping address in the network interface:

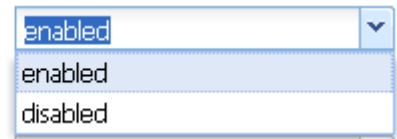
1. Expand the Configuration Manager slider > Global settings.



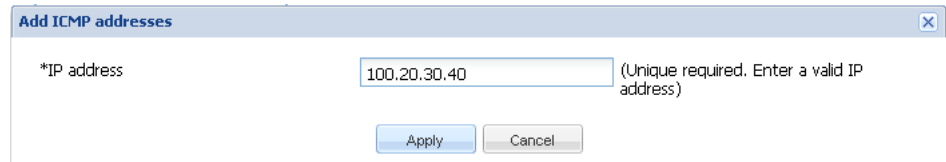
2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.

3. Click the row of the network interface you want to edit in the Network interface table and click **Edit**.
4. **ICMP**—Select enabled from the combination box.

ICMP



5. Scroll to the Icmp addresses table and click **Add**. The Add Icmp address dialog box appears.
6. **ICMP IP**—Choose an IP address that you want to use as ICMP ping addresses from the combination box.



7. Click **OK** to close the dialog box. The ICMP IP address appears in the Icmp addresses list.
8. Repeat steps 6 and 7 to add multiple ICMP IP addresses.
9. Click **Apply**.

SNMP

This section explains how to configure Simple Network Management Protocol (SNMP), trap receivers, and syslog servers. These features are not essential for baseline Net-Net SBC service, but they are necessary to use Acme Packet's Net-Net Central to manage Net-Net SBCs. They provide important monitoring and system health information that contribute to a robust deployment of the Net-Net system.

For detailed descriptions of the MIBs and information concerning their architecture and use, please refer to the *Net-Net MIB Reference Guide*.

Overview

SNMP is used to support monitoring of network-attached devices for conditions that warrant administrative attention. SNMP is comprised of three groups of settings on a Net-Net SBC. These settings are system-wide configurations including MIB contact information, SNMP community settings, and trap receivers.

Basic SNMP Parameters

The Net-Net SBC includes several parameters that control basic SNMP functionality. The MIB-related elements are for informational purposes, and are helpful if set. The remainder of the parameters determines if certain Net-Net SBC events are reported to the SNMP system.

SNMP Community

An SNMP community is a grouping of network devices and management stations used to define where information is sent and accepted. An SNMP device or agent might belong to more than one SNMP community. SNMP communities provide a type of password protection for viewing and setting management information within a community.

SNMP communities also include access level settings. They are used to define the access rights associated with a specific SNMP community. The Net-Net SBC lets you define two types of access levels: read-only and read-write. You can define multiple SNMP communities on a Net-Net SBC to segregate access modes per community and NMS host.

Trap Receivers

A trap receiver is an application used to receive, log, and view SNMP traps for monitoring the Net-Net SBC. An SNMP trap is the notification sent from a network device, the Net-Net SBC in this case, that declares a change in service. Multiple trap receivers can be defined on a Net-Net SBC either for redundancy or to segregate alarms with different severity levels to individual trap receivers.

Each server that Net-Net Central is installed on should be configured as a trap receiver on all Net-Net SBC's managed by Net-Net Central.

Configuring SNMP

This section describes how to configure your Net-Net SBC to work with external SNMP systems. Sample configurations are also provided.

SNMP Configuration Overview

1. Configure the SNMP identification information. This step includes configuring the MIB system contact, name, and location parameters.
2. Set the general SNMP parameters to enable or disable SNMP on your Net-Net SBC. Also included here are switches that govern how the SNMP system responds to specified events.
3. Set the syslog events (explained in the next section). They can trigger SNMP syslog traps. Parameters dealing with SNMP monitoring syslog events are configured here.
4. Set SNMP communities. Their configuration is separated into a unique configuration element.
5. Set trap receivers. Their configuration is separated into a unique configuration element.

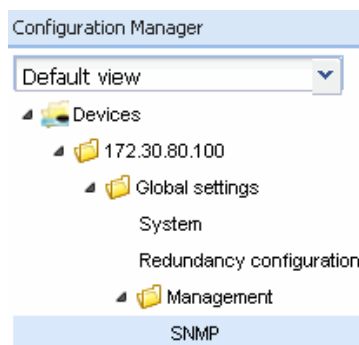
System Wide Configuration for SNMP

This section describes the system-wide SNMP parameters that set global SNMP information.

Configuring SNMP

This section describes how to configure SNMP.

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click SNMP. The SNMP data window appears in the content area.
3. **System name**—Enter the name of this Net-Net SBC presented within MIB transactions. This value, along with the target name of the Net-Net SBC (identified in the boot parameters) are the values reported for MIB-II when an SNMP GET is issued by the EMS for the MIB-II sysName variable. This parameter has no direct relation to the hostname parameter in the system configuration element.

By convention, this is the node's FQDN. For SNMP MIB-II sysName GETs, the Net-Net SBC returns SNMP communications in the following format:

`<targetName>[.<mib-system-name>]`

target Name is the value configured in the target name (tn) boot parameter and mib-system-name is the value configured in this field.

4. **System contact**—Enter the contact information used with in the Net-Net SBC's MIB transactions. The SNMP agent sends this information to an EMS in response to an SNMP Get for the MIB-II sysContact MIB variable. This parameter's value can be a textual identification of your company's contact person for the Net-Net SBC and/or information about how to contact that person.
5. **System location**—Enter the physical location of this Net-Net SBC that is reported within MIB transactions. This parameter is reported when an SNMP GET is issued by the EMS for the MIB-II sysLocation variable. This parameter has no direct relation to the location field in the system configuration element.

SNMP

System name	acme
System contact	QA
System location	QA lab

6. **State**—Choose the SNMP operational state from the combination box.

State	enabled enabled disabled

SNMP Servers

1. Click **Add** in the SNMP community section. The Add SNMP community dialog box appears.
2. **Community name**—Enter the SNMP community name of an active community where this Net-Net SBC can send or receive SNMP information. A community name value can also be used as a password to provide authentication, thereby limiting the NMSs that have access to this Net-Net system. With this field, the SNMP agent provides trivial authentication based on the community name that is exchanged in plain text SNMP messages.

The dialog box titled "Add instance" shows the configuration for a new SNMP community. It includes a label "systemConfig" and a section header "SNMPCOMMUNITY". The main field is labeled "*Community name" and contains the text "public1" with a note "(Unique required)". At the bottom are "Apply" and "Cancel" buttons.

3. Click **Apply**.
4. Select the community name from the table and click **Edit**.
5. **Access mode**—Choose the access mode from the combination box. The access level determines the permissions that other NMS hosts can wield over this Net-Net SBC. This parameter can be set to **read-only** and **read-write**.
 - **read-only**—allows GET requests
 - **read-write**—allows both GET and SET requests

6. Click **Add** in the Addresses table.

Addresses

172.30.0.30
10.0.200.5
172.30.10.129
172.30.80.10
10.0.200.63
10.1.20.149

Buttons: Add, Delete

7. **IP address**—Enter the IPv4 address that is valid within this SNMP community. This IPv4 address corresponds with the IPv4 address of NMS applications that monitor or configure this Net-Net SBC.

Add Addresses

*IP address: 172.30.10.129 (Unique required. Enter a valid IP address)

Buttons: Apply, Cancel

8. Click **Apply**.
9. Click **Apply** at the bottom of the content area.
10. Click **Add** in the Trap receiver table. The Add trap receiver dialog box appears.

Add Trap receiver

*IP address:port: 172.30.1.82:162 (Unique required. Enter IP and port in the format: x.x.x.x:xx)

*Community name: public

Buttons: Apply, Cancel

11. **IP addresses:port**—Enter the IP address and port where traps are sent in the format x.x.x.x:xx.
12. **Community name**—Enter the name of the community to which the trap-receiver belongs.
13. Click **Apply**.
14. Select the newly created trap receiver from the table and click **Edit**.
15. **Filter level**—Choose the filter level threshold that indicates the severity level at which a trap to be sent to this particular trap receiver.

Example: A trap with a severity level of Critical is generated, the SNMP agent will only send this trap to NMSs that are configured in a trap-receiver element and have a filter-level parameter of Critical.

The following table maps Syslog and SNMP alarms to trap receiver filter levels.

Filter Level	Syslog Severity Level	(SNMP) Alarm Severity Level
Critical	- Emergency (1) - Critical (2)	- Emergency - Critical
Major	- Emergency (1) - Critical (2) - Major (3)	- Emergency - Critical - Major
Minor	- Emergency (1) - Critical (2) - Major (3) - Minor (4)	- Emergency - Critical - Major - Minor
All	- Emergency (1) - Critical (2) - Major (3) - Minor (4) - Warning (5) - Notice (6) - Info (7) - Trace (8) - Debug (9)	- Emergency - Critical - Major - Minor - Warning

When configuring the trap-receiver element for use with Net-Net Central systems, Acme Packet recommends that the filter-level parameter be set to **All** for that configuration element that includes Net-Net Central servers.

16. Click **Apply**.
17. Click **Back** to continue configuring SNMP.

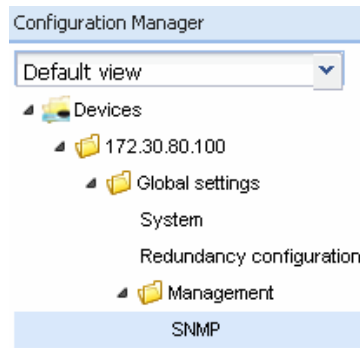
Traps

1. **Authentication traps**—Choose whether the SNMP authentication traps are enabled from the combination box. If an SNMP request fails authentication because of an IPv4 address and SNMP community mismatch, the SNMP request will be rejected. This field determines if an SNMP trap will be sent in response to the authentication failure.
2. **Unique TRAP identifiers**—Choose the state of unique trap IDs per syslog event functionality from the combination box. When this parameter is enabled, the Net-Net SBC generates traps with unique trap-IDs for each syslog event. If this parameter is set to disabled, a single trap-ID is used for all events, with different values in the description string.
3. **Environment monitoring traps**—Choose the state of SNMP environment monitor traps from the combination box. Environment traps include main board PROM temperature, CPU voltage, power supplies, fan speeds, etc.

Authentication traps	enabled	▼
Unique trap identifiers	enabled	▼
Environment monitoring traps	enabled	▼

MIB Transaction Contact

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click SNMP. The SNMP data window appears in the content area.
3. **System contact**—Enter the contact information used within the Net-Net system's MIB transactions. The SNMP agent sends this information to an NMS in response to an SNMP Get for the MIB-II sysContact MIB variable. This parameter's value can be a textual identification of your company's contact person for the Net-Net system and/or information about how to contact that person.

Note: There are no Net-Net Central equivalents for the mib-system-name and mib-system-location ACLI parameters.

4. Click **Apply** at the bottom of the window to complete the SNMP configuration.

Alarm Thresholds

You can configure alarm thresholds for the following Net-Net SBC characteristics:

- CPU utilization (as a percentage of total CPU capacity)
- memory utilization (as a percentage of total memory capacity)
- license utilization (as a percentage of licensed capacity)

You can now also configure the following types of thresholds at which the Net-Net SBC sends a SNMP trap:

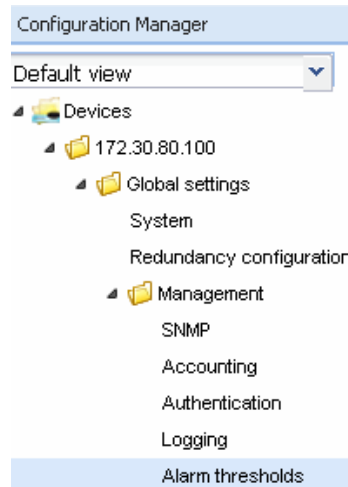
- critical
- major
- minor

Configuring Alarm Thresholds

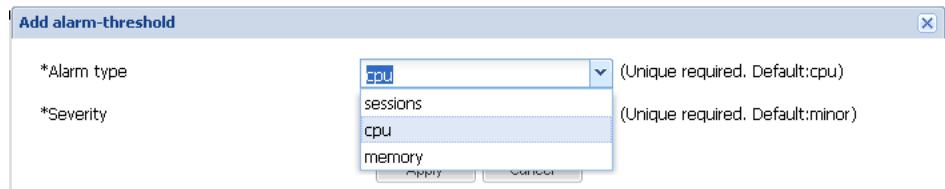
You can configure alarm thresholds using Net-Net Central.

To configure alarm thresholds:

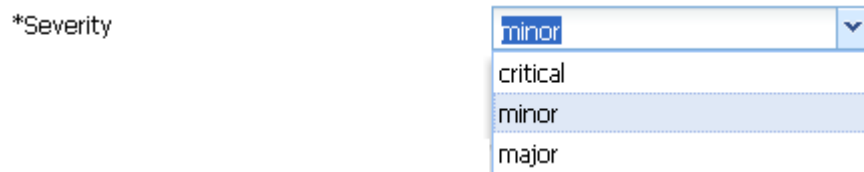
1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Alarm thresholds. The Alarm threshold table appears in the content area.
3. Click **Add**. The Add alarm-threshold dialog box appears.
4. **Alarm Type**—Choose the alarm threshold type from the combination box.



5. **Severity**—Choose the level of severity from the combination box.



6. Click **Apply**.
7. Select the newly created Alarm threshold from the table and click **Edit**.
8. **Value**—Enter the value at which, if exceeded, an alarm is generated. This value is expressed as a percentage ranging from 1 to 99.
9. Click **Apply**.
10. Click **Apply**.

ETC Monitoring

The Net-Net SBC can monitor an Enhanced Traffic Control (ETC) NIU card's data cores' average load and memory usage. Any excessive load is reported as an alarm, independently for CPU load and memory usage. When data core or memory usage rises above a set threshold, an alarm is enabled and a trap is sent. No new calls are accepted by the system until the alarm is cleared.

ETC CPU and Memory Monitoring

The Net-Net SBC uses the existing host CPU usage alarm threshold for monitoring the ETC card's data core usage on rolling 5 second windows. The Net-Net SBC monitors ETC card memory similarly to data core load with the existing host memory usage alarm threshold.

The default threshold for both the CPU and Memory Usage alarms is 90%. A CPU load at 90% triggers a minor alarm, while the Memory reaching 90% triggers a critical alarm. These values are changed by first setting the **type** parameter to **cpu** in the *alarm threshold* configuration element, and setting the **value** and **severity** parameters appropriately. Next the **type** parameter is set to **memory** and you set the **value** and **severity** parameters appropriately. These two *alarm threshold* objects are saved.

If the ETC card's data core or memory usage rises above the thresholds, independent alarms are reported, independent traps are sent, and no new calls are accepted by the system until the alarms are cleared.

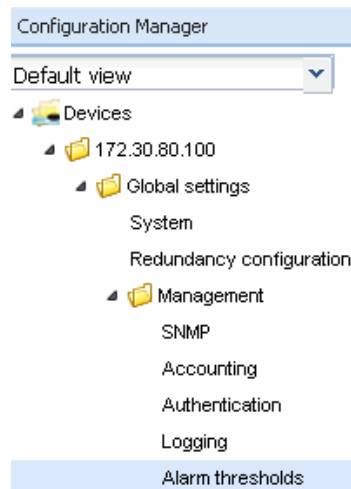
The ETC alarms are cleared after 10 consecutive seconds of the over-threshold reading dropping below the configured **value**. When the alarms are cleared, a clear trap is sent and new, incoming calls are accepted again.

There is no health reduction for any alarm level to avoid system reboot or switch over.

Configuration Instructions

To configure data core and memory usage thresholds applicable to the ETC NIU card (and the host):

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Alarm thresholds. The Alarm threshold table appears in the content area.
3. Click **Add**.
4. **Severity**—Select the severity for this alarm.
5. Scroll down the content manager to the Messages section.
6. **Type**—Select one of the following values:
 - **cpu**—Select to add a TCA for CPU load. This value applies to both the host CPU and the ETC data cores.

- **memory**—Select to add a TCA for memory usage load. This value applies to both the host memory and the ETC memory banks
7. Click **Apply**.
 8. Select the alarm threshold from the table and click **Edit**.
 9. **Value**—Enter the percentage load seen on the data cores that trigger this alarm.
 10. Click **Apply**.

Syslog and Process Logs

Logging events is a critical part of diagnosing misconfigurations and optimizing operations. Net-Net SBCs can send both syslog and process log data to appropriate hosts for storage and analysis.

Overview

The Net-Net SBC generates two types of logs, syslogs and process logs. Syslogs conform to the standard used for logging servers and processes as defined in RFC 3164.

Process logs are Acme Packet proprietary logs. Process logs are generated on a per-task basis and are used mainly for debugging purposes. Because process logs are more data inclusive than syslogs, their contents usually encompass syslog log data. A special application must be run on a remote server to receive process logs. Contact your Acme Packet sales representative directly or through email at support@acmepacket.com for more information about the process log application.

Syslog and process log servers are both identified by an IPv4 address and port pair.

Process Log Messages

Process log messages are sent as UDP packets in the following format:

```
<file-name>:<log-message>
```

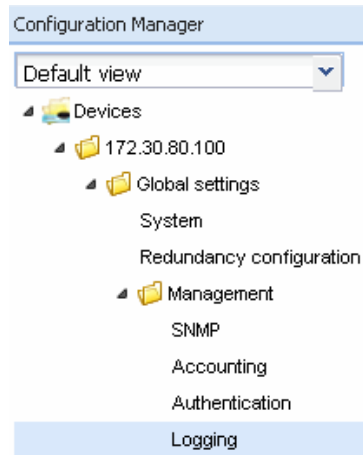
In this format, <filename> indicates the log filename and <log-message> indicates the full text of the log message as it would appear if it were written to the normal log file.

Configuring Syslog and Process Logs

This section describes how to configure syslog and process log information.

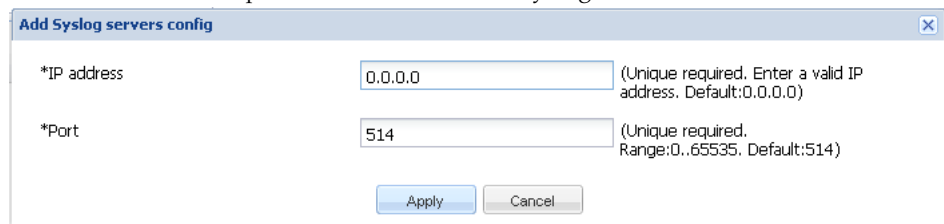
Syslog

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Logging. The Logging data appears in the content area.
3. **Log level**—Choose the syslog severity level that causes syslog traps to be sent to an NMS from the combination box. When this criteria is met and the appropriate SNMP trap is sent, an entry is written to the SNMP Syslog History Table.
4. **SNMP syslog notify traps**—Choose whether SNMP traps are sent when a Net-Net system generates a syslog message from the combination box. The SNMP agent will send a trap when a syslog is generated if the following conditions are met:
 - SNMP is enabled.
 - This field is enabled.
 - The syslog severity level is equal to or greater than the severity level configured in the SNMP Syslog Level field.
5. **Syslog notify traps-history table length**—Enter the length of syslog trap history table. the length of the syslog trap history table. When a syslog message that meets the SNMP syslog level field criteria is generated and SNMP is enabled, the SNMP agent adds that message to a history table. This parameter indicates the number of entries the table can contain. The syslog history table can range from 1 to 500 entries. By default, the parameter is set to 1. Once the last table entry is filled, the oldest entry will be overwritten with a new entry.
6. Scroll to the Syslog servers config table and click **Add**. The Add Syslog servers config dialog box appears.
7. **IP address**—Enter the IP address for the new syslog server.

8. **Port**—Enter the port number of the new syslog server.



The dialog box titled "Add Syslog servers config" contains two input fields. The first field is labeled "*IP address" and contains the text "0.0.0.0". To its right is a tooltip that reads "(Unique required. Enter a valid IP address. Default:0.0.0.0)". The second field is labeled "*Port" and contains the text "514". To its right is a tooltip that reads "(Unique required. Range:0..65535. Default:514)". At the bottom of the dialog are two buttons: "Apply" and "Cancel".

9. Click **Apply**.
10. Select the Syslog server from the table and click **Edit**.
11. **Facility numerical code(RFC 3164) used for filtering by syslog servers**—Enter the facility numerical code. This is used to filter syslog servers based on RFC 3164.

Facility numerical code(RFC 3164) used for filtering by syslog servers

12. Click **Apply**.
13. Click **Back** to continue configuring Logs.

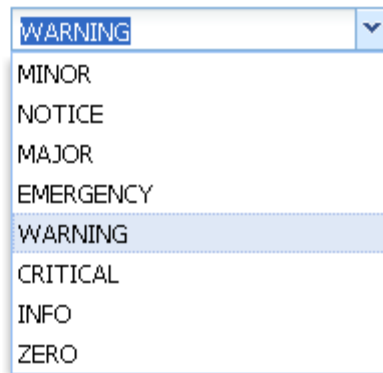
Once the last table entry is filled, the oldest entry will be overwritten with a new entry.

Global Log Level

14. **Log level**—Select the global log level for the system from the combination box in the System log level section of the screen; acmelog and syslog receive any messages that are at this level or higher. Valid values are: **ZERO**, **EMERGENCY**, **CRITICAL**, **MAJOR**, **MINOR**, **WARNING**, **NOTICE**, **INFO**. The default is **WARNING**.

System log level

Log level



A dropdown menu for selecting the log level. The current selection is "WARNING". The menu is open, showing a list of options: MINOR, NOTICE, MAJOR, EMERGENCY, WARNING (highlighted), CRITICAL, INFO, and ZERO.

15. Click **Apply**.

Configuring a Process Log Server

This section explains how to configure a process log server on your Net-Net SBC using your Net-Net EMS.

1. **Log level**—Select from the Log level combination box the starting log level that all processes running on the Net-Net SBC use. Each process running on the system has its own process log.
2. **Log mode**—Select the process log mode that you want to use from the combination box. The available choices are:
 - **REMOTE**—Process logs sent to the server only
 - **LOCAL**—Process logs sent to the file only
 - **BOTH**—Process logs sent to remote and local
 - **NONE**—No logs are sent
3. **IP address**—Set the IP address of the process log server. The default value for this parameter is 0. This parameter is only configurable when **REMOTE** or **BOTH** is selected for the log mode. The field becomes disabled when either **LOCAL** or **NONE** is selected.
4. **IP port**—Set the port number to use for the process log server. The default value for this parameter is 0.

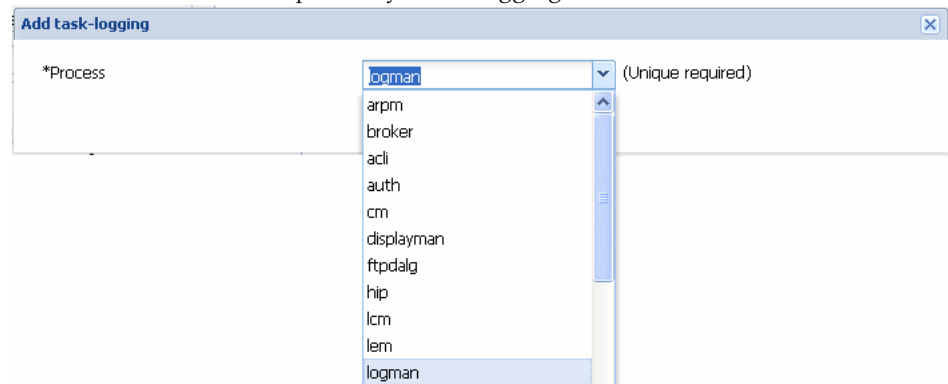
Process log

Log level	NOTICE ▼
Log Mode	LOCAL ▼
IP address	0.0.0.0
IP port	0

Task-Specific Logging

To set a log level for a specific task:

1. Click **Add** in the Task logging table. The Add Task logging dialog box appears.
2. **Process**—Select the process you are logging from the combination box.



3. Click **Apply**.
4. Select the Task log from the table and click **Edit**.
5. **Level**—Select the log level at which the task will write event messages. If configured, this value supersedes the global RTC startup logging level, defined by the process log level parameter.

6. **Mode**—Enter the logging mode you want to use: remote, local, both, or none.
7. **Log address**—Enter the log address and port to which you are writing.

*Process	broker
Level	WARNING
Mode	LOCAL
Log address	120.10.20.34:15

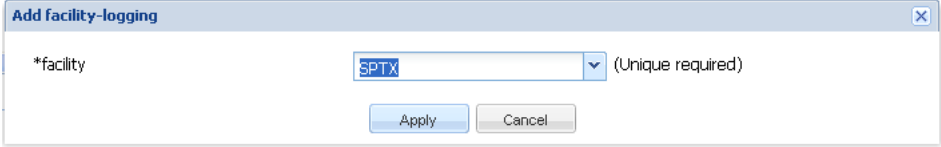
8. Click **Apply**.

Facility Logging for Specific Tasks

You can also configure the Net-Net SBC to perform facility-specific logging for specified tasks.

A log type facility is a second filtering criteria used when writing to log files. The facility is a tag that describes the context of the event. Facility messages are filtered by the task and then by the mechanism that writes data to task log files to limit the amount of information that gets written into a task's log file.

1. Select the task log from the Task logging config table that you want to configure facility logging for and click **Edit**. The Edit Task logging config window appears.
2. Click Add in the Facility logging config section of the screen. The Add facility-logging dialog box appears.
3. **Facility**—Select the facility type message you want to be logged.



The dialog box titled "Add facility-logging" contains a dropdown menu for "*facility" with "SPTX" selected. To the right of the dropdown is the text "(Unique required)". At the bottom of the dialog are "Apply" and "Cancel" buttons.

4. Click **Apply**.
5. Select the facility logging configuration from the table and click **Edit**.
6. **Level**—Select the log level from the combination box.
7. **Mode**—Select the mode from the combination box.

8. **Log address**—Enter the remote log server address and port; your entries need to follow the IPaddress:port format.

Level	WARNING ▼
Mode	LOCAL ▼
Log address	120.10.20.34:15

9. Click **Apply**.
10. Click **Apply** to complete syslog and process log configuration.

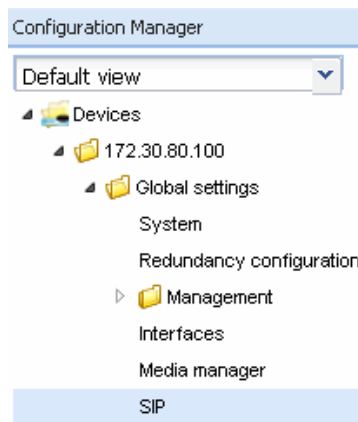
Message Rate Statistics

The Net-Net SBC provides message rate statistics for SIP, DNS, and ENUM traffic.

SIP Message Rate Statistics

To enable full SIP message rate statistics:

1. Expand the Configuration Manager slider > Global settings.

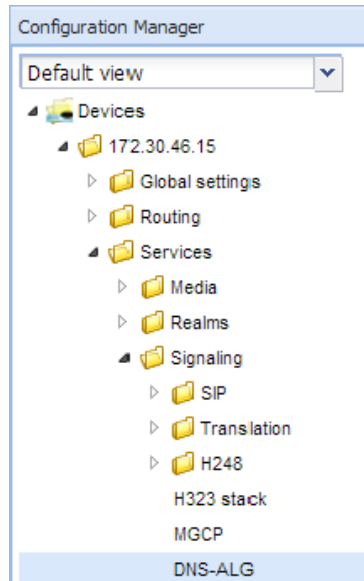


2. Click SIP.
3. Select a SIP configuration and click **Edit**.
4. **Maintain extra method statistics**—Set this parameter to enabled for the Net-Net SBC to collect and track SIP method statistics per second.
5. Click **Apply**.

DNS ALG Message Rate Statistics

To enable full DNS ALG message rate statistics:

1. Expand the Configuration Manager slider > Services > Signaling.

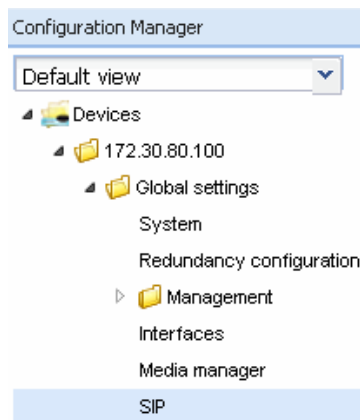


2. Click DNS-ALG.
3. Select a DNS-ALG configuration from the table and click **Edit**.
4. **Extra DNS-ALG stats**—Set this parameter to enabled for the Net-Net SBC to collect message rate statistics for DNS ALG objects.
5. Click **Apply**.

ENUM Server Message Rate Statistics

To enable ENUM message rate statistics:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Select a SIP configuration and click **Edit**.
4. **Maintain extra ENUM statistics**—Set this parameter to enabled for the Net-Net SBC to collect and track ENUM message statistics per second.

5. Click **Apply**.

Host Routes

This section explains how to configure host route exceptions on the Net-Net SBC.

Overview

Host routes let you insert entries into the Net-Net SBC's routing table. These routes affect traffic that originates at the Net-Net SBC's host process. Host routes are used primarily for steering management traffic to the correct network.

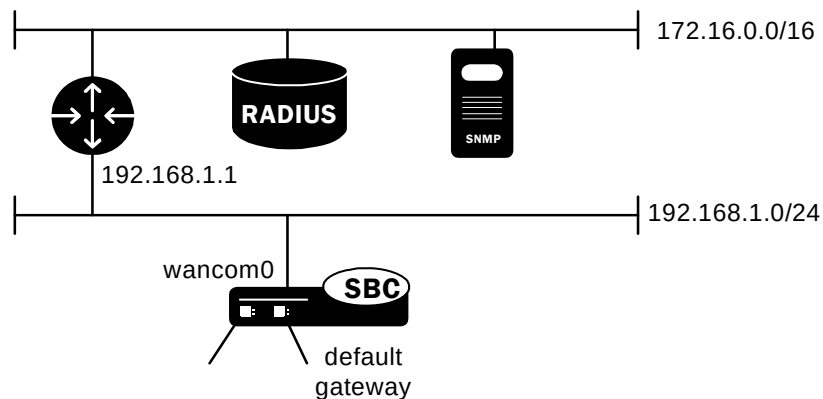
When traffic is destined for a network that is not explicitly defined on a Net-Net SBC, the default gateway (located in the system config) is used. If you try to route traffic to a specific destination that is not accessible through the default gateway, you need to add a host route. Host routes can be thought of as a default gateway override.

Certain SIP configurations require that the default gateway is located on a front media interface. In this scenario, if management applications are located on a network connected to a rear-interface network, you will need to add a host route for management connectivity.

When source-based routing is used, the default gateway must exist on a front media interface. Host routes might be needed to reach management applications connected to a wancom port in this kind of situation as well.

Host Routes Example

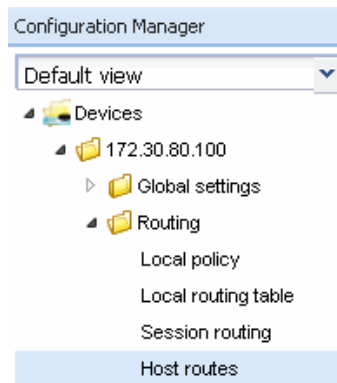
Because SIP signaling over media interfaces is enabled, the default gateway uses an IPv4 address assigned to a front interface. Maintenance services (SNMP and Radius) are located on a network connected to, but separate from, the 192.168.1.0/24 network on wancom0. In order to route Radius or SNMP traffic to an NMS (labeled as SNMP in the following example), a host route entry must be a part of the Net-Net SBC configuration. The host route tells the host how to reach the 172.16.0.0/16 network. The actual configuration is shown in the example in the next section of this guide.



Configuring Host Routes

This section describes how to configure host routes.

1. Expand the Configuration Manager slider > Routing.



2. Click Host routes. The Host routes table opens in the content area.
3. In the Host routes table, click **Add**.
4. **IP address**—Enter the IP address of the destination network that this host route points toward.
5. **Subnet mask**—Enter the netmask portion of the destination network for the route you are creating.
6. **Gateway IP address to destination network**—Enter the IP address of the gateway that traffic destined for the address defined in the first two elements should use as its first hop.

The 'Add Host routes' dialog box is shown. It has three input fields with the following values and tooltips:

- *IP address: 110.20.30.40 (Unique required. Enter a valid IP address)
- *Gateway IP address to destination network: 116.37.18.10 (Enter a valid IP address)
- *Subnet mask: 100.14.13.27 (Enter a valid IP address)

At the bottom, there are 'Apply' and 'Cancel' buttons.

7. Click **Apply** to continue. The new host route appears in the Host route table.

Editing Host Routes

1. Select a host route in the Host route table and click Edit.
2. **Subnet mask**—Edit the netmask portion of the destination network for the route you are creating.
3. **Gateway IP address to destination network**—Edit the IP address of the gateway that traffic destined for the address defined in the first two elements should use as its first hop.
4. **Description**—Enter an optional descriptive string to identify the configuration and its function.
5. Click **Apply**.

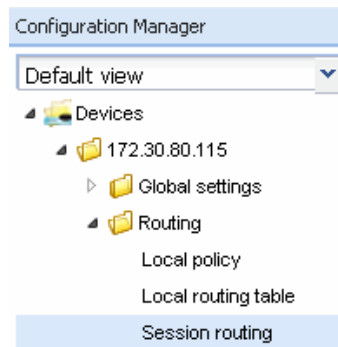
Holidays

You can define holidays that the Net-Net SBC recognizes. Holidays are used to identify a class of days on which a local policy is enacted. Because holidays are entered on a one-time basis per year, you must configure a new set of holidays yearly.

Configuring Holidays

To configure holidays:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. In the Holiday dates and description table, click **Add**.
5. **Date**—Enter the holiday's date in YYYY-MM-DD format. You can access a calendar by clicking the button next to the textbox.
6. **Description**—Enter a short description for the holiday you are configuring. If the description contains words separated by spaces, enter the full description surrounded by quotation marks.

The 'Add instance' dialog box for 'sessionRouter' is shown. It contains the following fields and controls:

- SESSIONROUTER/HOLIDAY** (Section Header)
- *Description**: Textbox containing '2006-12-25' with a calendar icon. To the right, it says '(Unique required)'.
- *Date**: Textbox containing 'Christmas' with a calendar icon. To the right, it says '(Unique required)'.
- Configuration Guidelines**: A link with a dropdown arrow.
- Buttons**: 'Apply' and 'Cancel' buttons at the bottom.

7. Click **Apply** to continue. The new holiday appears in the Holiday Dates & Descriptions list.
8. Click **Apply** to complete configuring holidays.

Enhanced Control of UDP and TCP Ports

This section explains how to configure the Net-Net SBC for finer control of the set of UDP and TCP ports that on which the Net-Net SBC provides services. The settings you can configure have an impact on:

- UDP/TCP port 111 (the RPC services port), which is disabled on Net-Net SBC startup but can be enabled in the boot parameters
- TCP ports 3000 (used when notify commands are issued remotely, i.e. via the Net-Net Central) and 3001 (used for remote configuration, i.e. via the Net-Net Central), which can now be enabled or disabled in the system configuration

Neither configuration for these features is covered by RTC, so you must reboot your Net-Net SBC for changes to take effect. Be aware that rebooting can cause system downtime, and plan accordingly.

ACLI Instructions and Examples

To enable port 111 using Net-Net SBC boot parameters:

1. In Superuser mode, type **configure terminal** and press <Enter>
ACMEPACKET# **configure terminal**
2. To enter the boot parameters so that you can configure them, type **bootparam** and press <Enter>.
ACMEPACKET(configure)# **bootparam**
3. Press <Enter> to scroll through the list of boot parameters until you reach the setting for flags.

To set this value correctly, you need to add the value 0x200000 to your existing flag setting in the boot parameters. In the example below, the existing flag value is 0x300008. When the value 0x200000 is added, the result is 0x230008. The result is the value that you need to set.

When you reach the **flag** setting, type the value representing the flags you need (0x230008 in the example below) and press <Enter>. Continue to press <Enter> to finish scrolling through the rest of the boot parameters.

'.' = clear field; '-' = go to previous field; ^D = quit

```
boot device          : wancom0
processor number : 0
host name : acmepacket8
file name : /tffs0/sd220p9.gz
inet on ethernet (e): 10.0.1.57:ffff0000
inet on backplane (b): 0.0.0.0
host inet (h)       : 10.0.1.5
gateway inet (g)    : 10.0.0.1
user (u)            : user
ftp password (pw)   : password
flags (f)           : 0x300008 0x230008
target name (tn)    : acmesystem
startup script (s)  : 0
other (o)           :
```

NOTE: These changed parameters will not go into effect until reboot. Also, be aware that some boot parameters may also be changed through the PHY and Network Interface Configurations.

ACMEPACKET(configure)#

4. Type **exit** to return to the main Superuser menu so that you can reboot your Net-Net SBC and apply the settings you have entered.

```
ACMEPACKET(configure)# exit
```

5. Reboot your Net-Net SBC. Type a **y** and press <Enter> to reboot.

```
ACMEPACKET# reboot
```

```
-----  
WARNING: you are about to reboot this SD!  
-----
```

```
Reboot this SD [y/n]?:y
```

To control TCP ports 3000 and 3001 in the system configuration:

1. In Superuser mode, type **configure terminal** and press <Enter>

```
ACMEPACKET# configure terminal
```

2. Type **system** and press <Enter>.

```
ACMEPACKET(configure)# system
```

3. To set parameters in the system configuration, type **system-config** and press <Enter>.

```
ACMEPACKET(system)# system-config
```

4. To start editing the system configuration, type **select** and press <Enter>.

```
ACMEPACKET(system-config)# select
```

5. The parameter controlling ports 3000 and 3001 is called **remote-control**, and its default is enabled. To disable the ports, set this parameter to disabled.

```
ACMEPACKET(system-config)# remote-control disabled
```

6. Save your changes and exit the system configuration and man system menus.

```
ACMEPACKET(system-config)# done
```

```
ACMEPACKET(system)# exit
```

7. Type **exit** to return to the main Superuser menu so that you can reboot your Net-Net SBC and apply the settings you have entered.

```
ACMEPACKET(configure)# exit
```

8. Reboot your Net-Net SBC. Type a **y** and press <Enter> to reboot.

```
ACMEPACKET# reboot
```

```
-----  
WARNING: you are about to reboot this SD!  
-----
```

```
Reboot this SD [y/n]?:y
```

HDR Trap for Data Push Success

When you use historical data recording (HDR) on your Net-Net SBC, you can configure the system to send a trap confirming successful data pushes to designated network elements. This trap has a default level of MINOR. It contains the name of the node that successfully pushed the HDR file to an HDR server, a unique file name for the HDR file that was pushed, and the IP address of the push receiver (configured in the global collection configuration).

If you enable the trap without any configured HDR groups, the Net-Net SBC sends the trap for all HDR groups. Refer to the *System Configuration* chapter of the *Net-Net Configuration Guide* for a complete list of HDR group record types. If you configure specific groups, then the system sends the trap only for those groups.

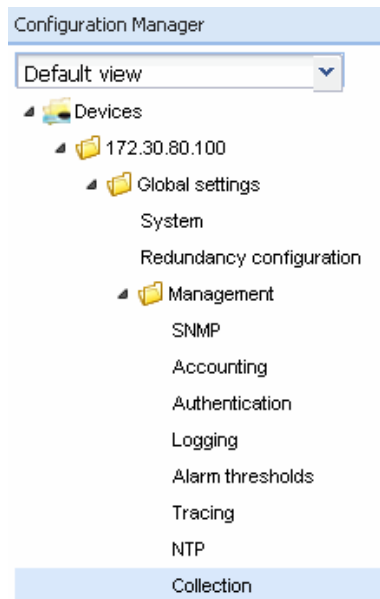
Note: You should consider the system performance impact of enabling this trap before you do so.

Configuring HDR Data Push Success Trap

You enable the HDR data push success trap in the global collection settings, along with the sampling interval, push interval, and other settings.

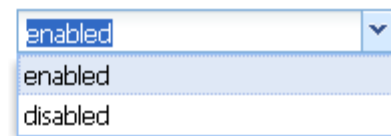
To enable the HDR trap for data push success:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Collection.
3. **Generate trap for successful pushes**—Select enabled from the combination box to enable the Net-Net SBC to send a trap confirming successful data pushes to HDR servers.

*Generate trap for successful pushes



4. Click **Apply**.

DNS Transaction Timeout

This section explains how to configure the DNS transaction timeout interval on a per network-interface basis. You can currently configure the Net-Net SBC with a primary and two optional backup DNS servers. The Net-Net SBC queries the primary DNS server and upon not receiving a response within the configured number of seconds, queries the backup1 DNS server and if that times out as well, then contacts the backup2 DNS server.

Retransmission Logic

The retransmission of DNS queries is controlled by three timers. These timers are derived from the configured DNS timeout value and from underlying logic that the minimum allowed retransmission interval should be 250 milliseconds; and that the Net-Net SBC should retransmit 3 times before timing out to give the server a chance to respond.

- Init-timer is the initial retransmission interval. If a response to a query is not received within this interval, the query is retransmitted. To safeguard from performance degradation, the minimum value allowed for this timer is 250 milliseconds.
- Max-timer is the maximum retransmission interval. The interval is doubled after every retransmission. If the resulting retransmission interval is greater than the value of max-timer, it is set to the max-timer value.
- Expire-timer: is the query expiration timer. If a response is not received for a query and its retransmissions within this interval, the server will be considered non-responsive and the next server in the list will be tried.

The following examples show different timeout values and the corresponding timers derived from them.

```

timeout >= 3 seconds
Init-timer = Timeout/11
Max-Timer = 4 * Init-timer
Expire-Timer = Timeout

timeout = 1 second
Init-Timer = 250 ms
Max-Timer = 250 ms
Expire-Timer = 1 sec

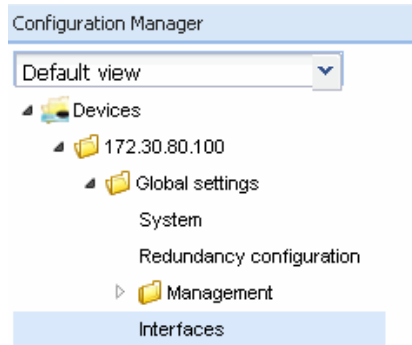
timeout = 2 seconds
Init-Timer = 250 ms
Max-Timer = 650 ms
Expire-Timer = 2sec

```

Configuring DNS Transaction Timeout

To configure DNS transaction timeout:

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.
3. Select a network interface in the Network interface table and click **Edit**.
4. Scroll to the DNS section.

5. **DNS timeout**—Enter the total time in seconds you want to elapse before a query (and its retransmissions) sent to a DNS server would timeout.

If a query sent to the primary DNS server times out, the backup1 DNS server is queried. If the query times out after the same period of time elapses, the query continues on to the backup2 DNS server.

DNS timeout

11

6. Click **Apply**.

Persistent Protocol Tracing

This section explains how to configure persistent protocol tracing to capture specific SIP and MGCP protocol message logs and persistently send them off the Net-Net SBC, even after rebooting the system.

About Persistent Protocol Tracing

You can configure sending protocol message logs off of the Net-Net SBC, and have that persist after a reboot. You no longer have to manually issue the notify command each time you reboot.

To support persistent protocol tracing, you configure the following parameters:

- Protocol message tracing: enable/disable protocol message tracing (currently only sipmsg.log and alg.log) regardless of the process-log-level setting. If the process log level is set to trace or debug, protocol message tracing will not disable.
- Internal ACP message tracing: enable/disable internal ACP message tracing for all processes, regardless of process-log-level setting. This applies to all *.log (internal ACP message exchange) files other than sipmsg.log and alg.log. If the process log level is set to trace or debug, internal ACP message tracing will not disable.

- **Log filter:** determine what combination of protocol traces and logs are sent to the log server defined by the process log ip parameter value. You can set this parameter to any of the following values: **none**, **traces**, **logs**, or **all**.

The Net-Net SBC uses the value of this parameter in conjunction with the process log IP and process log port values to determine what information to send. If you have configured the process log IP and process log port parameters, choosing **traces** sends just the trace information (provided they are turned on), **logs** sends only process logs (log.*), and **all** sends everything (which is the default).

About the Logs

When you configure persistent protocol tracing, you affect the following types of logs. See the *Net-Net Maintenance and Troubleshooting Guide* for more details about all Net-Net SBC logs.

Note: Enabling logs can have an impact on Net-Net SBC performance.

Process Logs

Events are logged to a process log flow from tasks and are specific to a single process running on the Net-Net SBC. By default they are placed into individual files associated with each process with the following name format:

`log.<taskname>`

By setting the new log-filter parameter, you can have the logs sent to a remote log server (if configured). If you set log-filter to **logs** or **all**, the logs are sent to the log server. Otherwise, the logs are still captured at the level the process-log-level parameter is set to, but the results are stored on the Net-Net SBC's local storage.

Communication Logs

These are the communication logs between processes and system management. The logs are usually named `<name>.log`, with `<name>` being the process name. For example, `sipd.log`. This class of log is configured by the internal trace parameter.

Protocol Trace Logs

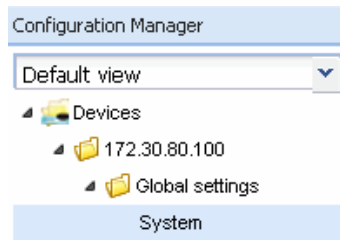
The only protocol trace logs included at this time are `sipmsg.log` for SIP and `alg.log` for MGCP. (The H.323 system tracing is not currently included.) All of the logs enabled with the call-trace parameter are sent to remote log servers, if you also set the log-filter parameter to **logs** or **all**.

Configuring Persistent Protocol Tracing

Before you configure persistent protocol tracing, ensure you have configured the process logs by setting the system configuration's process log IP parameter. See the *Net-Net Configuration Guide* and the *Net-Net Maintenance and Troubleshooting Guide* for more information.

To configure persistent protocol tracing:

1. Expand the Configuration Manager slider > Global settings.



2. Click System. The System window appears in the content area.
3. **Call trace**—Choose enabled from the combination box to enable protocol message tracing for sipmsg.log for SIP and alg.log for MGCP.



4. Click Apply.

Accounting

This section explains SIP accounting using the Remote Authentication Dial In User Service (RADIUS) Accounting System (RAS).

The Net-Net SBC uses the RADIUS protocol to send accounting messages. These messages are transmitted to one of a predefined list of accounting servers using a predefined forwarding strategy. RAS provides a mechanism for temporarily storing session initiation and completion statistics, and for delivering these statistics to accounting servers located elsewhere in the network.

SIP Session Accounting

The RAS client can record SIP session activity based on configuration and a CDR. The CDR determines which messages are generated and determines the RADIUS attributes included in the messages. The RAS client must be capable of sending CDRs to any number of RADIUS accounting servers, using the defined hunt, failover, round robin, fewest pending, or fastest server strategies.

The establishment, failed establishment, change, or removal of a SIP session can trigger RADIUS Accounting Request messages. The RAS might also send notification of its status (enabled/disabled). RADIUS Accounting Request messages include the following:

- **Start**—SIP session has started.
- **Interim-Update**—SIP session parameters have changed.
- **Stop**—SIP session has ended.
- **Accounting-On**—Creation of a new RADIUS client.
- **Accounting-Off**—RADIUS client has shut down.

Each SIP session might generate Start, Interim-Update, and Stop messages based on the local configuration when the session is initiated. Each Start message tells the

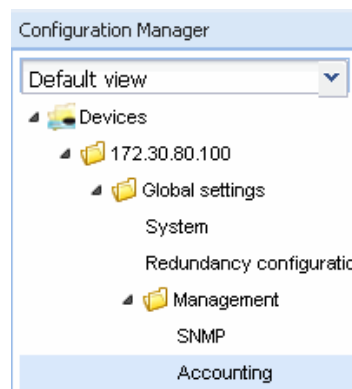
RADIUS server that a SIP session has started. Each Interim-Update message changes the SIP session parameters, and may report the session characteristics for the SIP session to that point. Each Stop message informs the RADIUS server that a SIP session has ended and reports session characteristics.

The RAS can transmit all RADIUS messages related to a SIP session at the end of the session, regardless of which messages are generated and when they are generated. You might choose this option to reduce the likelihood of the RADIUS messages being logged to different servers, or in different log files on the same server.

The RAS always generates a RADIUS Stop message when the SIP session ends, regardless of the SIP session termination cause. The termination cause and the session characteristics are reported.

Accounting Configuration

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. **State**—Select the State of the account configuration from the combination box.
5. **Host name**—Enter the Host name of the account configuration.
6. **Port**—Enter the port number of the account configuration.
7. **Start**—Select the type of SIP event that triggers the Net-Net system to transmit a RADIUS Start message from the combination box.

State	enabled
Host name	LOCALHOST
Port	1813
Protocol	RADIUS
Messages	
Start	OK

8. **Interim**—Click **Add** in the Interim table. Select the type of SIP event that triggers the Net-Net system to transmit a RADIUS Interim message in the Combination box.
9. Click **Apply**.

10. **Transmission timing**—Select whether you want the Net-Net system to transmit accounting information only at the close of a session from the combination box.

Transmission timing

11. **Max time-message retransmit (secs)**—Enter the time, in seconds, the Net-Net system continues to send each accounting message.
12. **Max. message queue before failover (#)**—Enter the number of accounting messages to be held in the message waiting queue before a failover situation status is enacted.

Max time-message retransmit(sec)

Max. message queue before failover (#)

Account Server

This section describes how to configure accounting servers on your Net-Net system.

1. Expand the Configuration Manager slider > Global settings > Management.

2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Click the Radius accounting tab at the top of the content area.
5. **Host name or IP address**—Enter the name or IP address of the host associated with the account server.

6. **IP port**—Retain the default **1813** or enter the number of the DUP port associated with the account server to which RADIUS messages are sent.
7. Click **Apply**.
8. Select the server from the table and click **Edit**.
9. **State**—Select the state of the account server on the Net-Net system from the combination box. The default is enabled.
10. **Min. round trip time threshold (ms)**—Enter the minimum round trip time of an accounting message. The default is 250 milliseconds. If the fastest RTT is the strategy for the account configuration the value you enter here can be used to determine an order of preference (if all the configured account servers are responding in less than their minimum RTT).
11. **VSA bundling**—Select **enabled** from the combination box if you want the account server to bundle the VSAs within RADIUS accounting messages. Select **disabled** if you do not. (Bundling means including multiple VSAs within the vendor value portion of the message.)
12. **Max. time ACK (sec)**—Enter the length of time in seconds that you want the Net-Net system with pending accounting messages to wait when it has not received a valid response from the target account server. The default is **60** seconds.
13. **Min. time-reinstate delay (sec)**—Enter the length of time in seconds that you want the Net-Net system to wait before resending messages to a disabled account server. The default is **30** seconds.

State	enabled ▼
VSA bundling	enabled ▼
*Host name or IP address	BEDFORD1
*IP port	1813
Min. round trip time threshold (ms)	250
Max. time ACK (sec)	60
Min. time-reinstate delay (sec)	30
Priority	0

14. **Shared secret**—Enter the secret password from the account server to the client. Transactions between the client and the the RADIUS server are authenticated by the shared secret.
15. **NAS ID**—Enter the NAS ID. The account server uses this value to identify the Net-Net system for the transmittal of accounting messages.

Shared secret	•••••
NAS ID	4321

16. Click **Apply** to finish configuring the accounting server.

Packet Trace

Packet tracing lets the Net-Net SBC mirror any communication between two endpoints, or between itself and a specific endpoint. To accomplish this, the Net-Net SBC replicates the packets sent and received, and then sends them to a trace server that you designate. Using the trace server, you can display the packets on software protocol analyzer. Currently, the Net-Net SBC supports:

- One configurable trace server (on which you have installed your software protocol analyzer)
- Sixteen concurrent endpoint traces

Note: The information included here describes how packet tracing works. It also contains the information you need to use Net-Net Central to configure packet tracing and to use the Acme Command Line Interface (ACLI) to start and stop packet tracing.

How It Works

You configure a trace server on the Net-Net SBC so that it knows where to send the mirrored packets. Then the Net-Net SBC bases the trace on one of its internally configured IP addresses (such as one for a SIP interface or for an H.323 interface).

Note: Set up the capture filter “ip proto 4” in your software protocol analyzer if you only want to see the results of the Net-Net SBC packet trace(s).

Once the trace is initiated, the Net-Net SBC duplicates all packets sent to and from the endpoint identified by the IP address that are sent or received on the specified Net-Net SBC network interface.

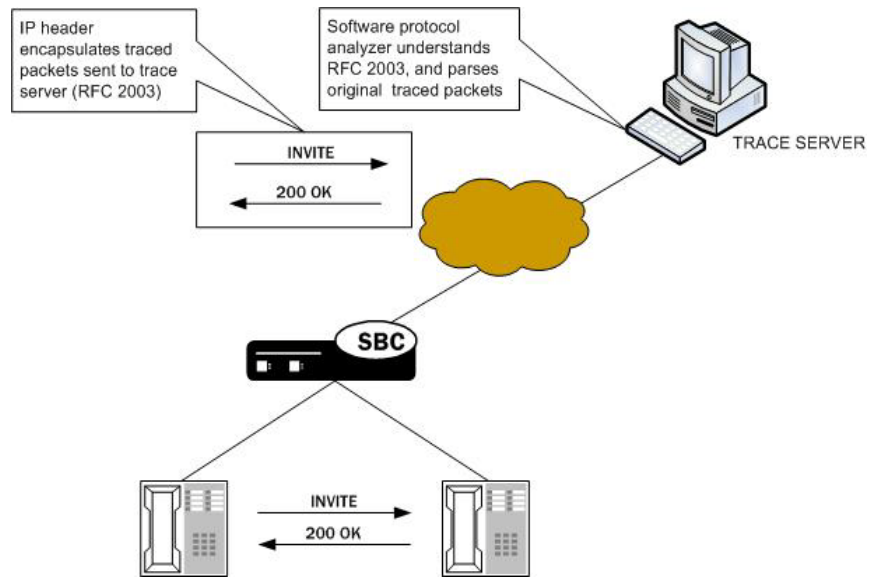
The Net-Net SBC then encapsulates the original packets in accordance with RFC 2003 (IP Encapsulation within IP); it adds the requisite headers, and the payload contains the original packet trace with the Layer 2 header removed. Because software protocol analyzers understand RFC 2003, they can easily parse the original traced packets. In order to see only packet traces information in your software protocol analyzer, you can use the capture filter “ip proto 4.”

It is possible that—for large frames—when the Net-Net SBC performs the steps to comply with RFC 2003 by adding the requisite header, the resulting packet might exceed Ethernet maximum transmission unit (MTU). This could result in packets being dropped by external network devices, but widespread support for jumbo frames should mitigate this possibility.

If the Net-Net SBC either receives or transmits IP fragments during a packet trace, then it will only trace the first fragment. The first fragment is likely to be a maximum-sized Ethernet frame.

The Net-Net SBC continues to conduct the packet trace and send the replicated information to the trace server until you instruct it to stop. You stop a packet trace with the ACLI **packet-trace stop** command. With this command, you can stop either an individual packet trace or all packet traces that the Net-Net SBC is currently conducting.

The following illustration shows the packet trace process.



Packet Trace Scenarios

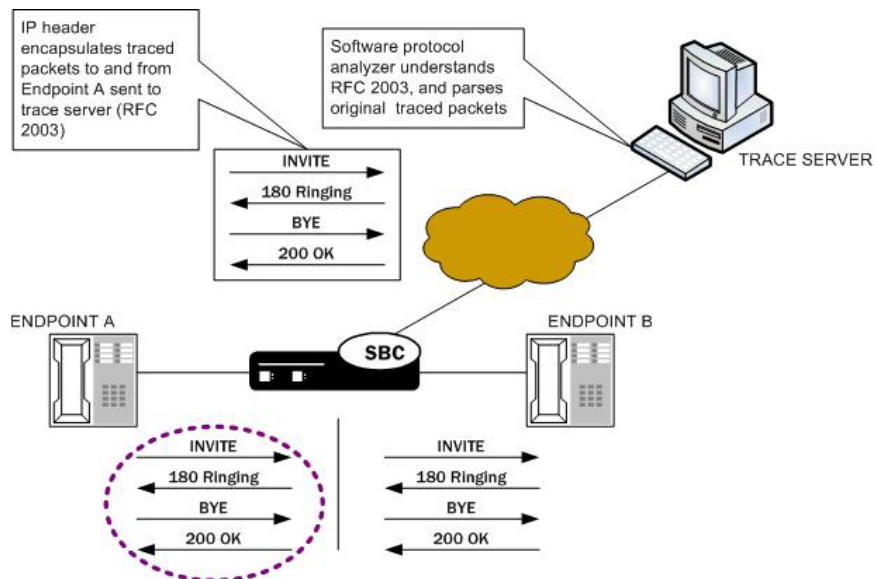
This section describes three possible ways that you might use the packet trace feature. You can examine communications sent to and from one endpoint, sent between two endpoints, or sent between ingress and/or egress Net-Net SBC interfaces to endpoints.

Packet Trace for One Endpoint

When you use the `packet-trace-state` command, the Net-Net SBC sets up packet tracing for one endpoint. The Net-Net SBC collects and replicates the packets to and from one endpoint. To enable this kind of trace, you set up one packet trace using the `packet-trace-start` command.

The commands you carry out would take the following form:

```
ACMEPACKET# packet-trace-start F01 <IP address of Endpoint A>
```



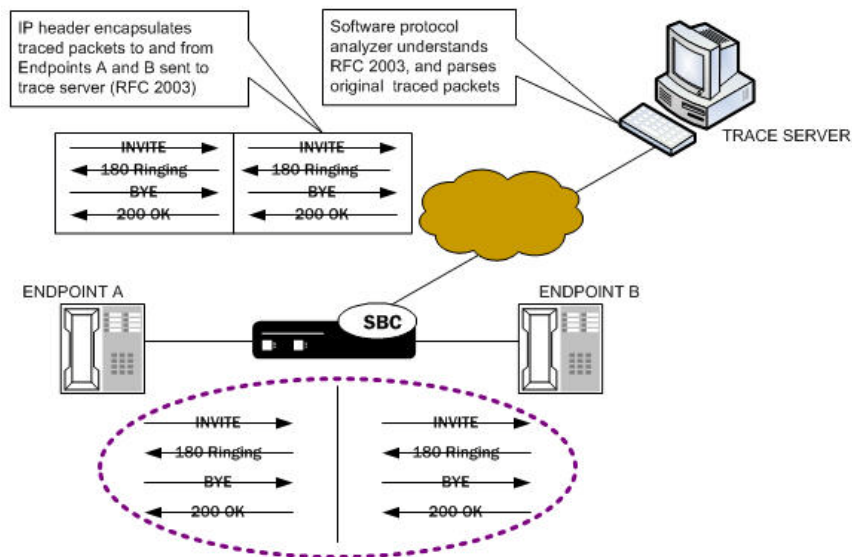
Packet Trace for Both Call Legs

If you want to trace both sides (both call legs), then you must set up individual traces for each endpoint—meaning that you would initiate two packet traces. The results of the trace will give you the communications both call legs for the communication exchanged between the endpoints you specify.

If you initiate a packet trace for both endpoints that captures both signaling and media, the signaling will be captured as usual. However, RTP will only be captured for the ingress call leg. This is because the Net-Net SBC performs NAT on the RTP, which means it cannot be captured on the egress call leg.

The commands you carry out would take the following form:

```
ACMEPACKET# packet-trace-start F01 <IP address of Endpoint A>
ACMEPACKET# packet-trace-start F02 <IP address of Endpoint B>
```

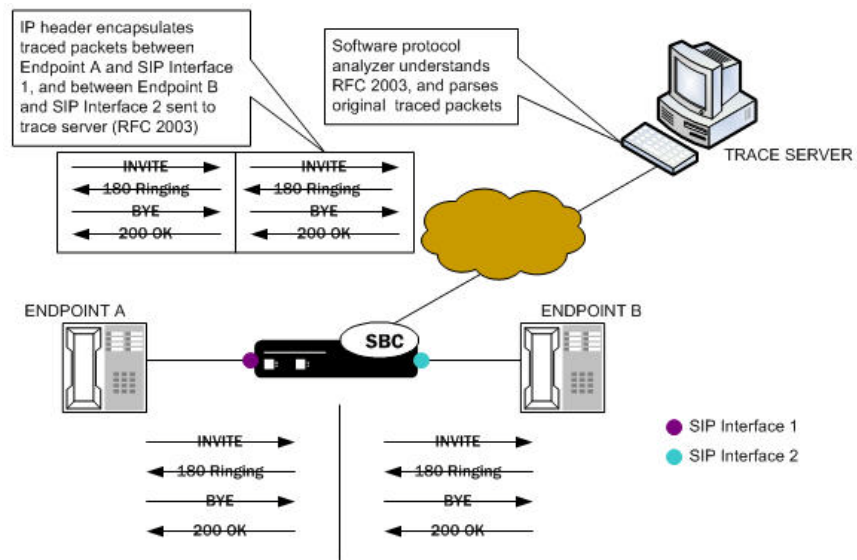


Packet Trace for a Net-Net SBC Signaling Address

You can perform a packet trace for addresses internal to the Net-Net SBC; this can be the address, for example, of a SIP or an H.323 interface. Using signaling interface addresses puts the emphasis on the Net-Net SBC rather than on the endpoints by allowing you to view traffic from specified interfaces.

The commands you carry out would take the following form:

```
ACMEPACKET# packet-trace-start F01 <IP address of Net-Net 4000
interface1>
ACMEPACKET# packet-trace-start F02 <IP address of Net-Net 4000
interface2>
```

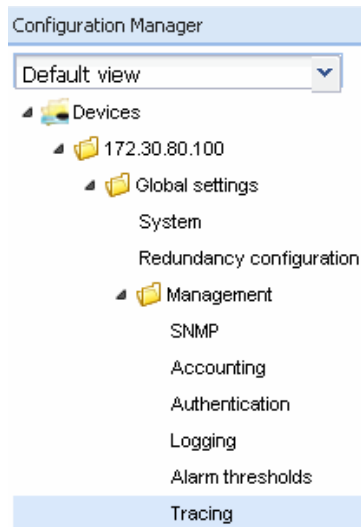



Configuring Packet Tracing

To configure packet tracing, you need to configure the Net-Net SBC with the trace server information so that the Net-Net SBC knows where to send replicated data. The Net-Net SBC supports one trace server.

To configure a trace server on your Net-Net SBC:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Tracing.
3. Click **Add** to add the Tracing configuration in the content area.

4. **Capture receiver state**—Select enabled from the combination box to activate the capture receiver to which you want to send the mirrored packets for calls you are packet-tracing.
5. **Trace receiver IP address**—Enter the IP address of the trace server.
6. **Network interface**—Click **Add**. The Add interfaces dialog box appears.
7. Choose the name of the Net-Net SBC network interface from which the Net-Net SBC is to send mirrored packets.
8. **Sub port**—Choose the sub port of the Net-Net SBC network interface from the combination box.

9. Click **Apply**. The network interface name and subport appear in the Network interface list.
10. Click **Apply**.

Alarm Synchronization

Two trap tables in the `ap-smgmt.mib` record trap information for any condition on the Net-Net SBC that triggers an alarm condition. You can poll these two tables from network management systems, OSS applications, and the Net-Net Central to view the fault status on one or more Net-Net SBCs.

There are two trap tables that support alarm synchronization, and by polling them you can obtain information about the current fault condition on the Net-Net SBC. These tables are:

- `apSysMgmtTrapTable`—You can poll this table to obtain a summary of the Net-Net SBC's current fault conditions. The table records multiples of the same trap type that have occurred within a second of one another and have different information. Each table entry contains the following:
 - Trap identifier
 - System time (synchronized with an NTP server)
 - `sysUpTime`
 - Instance number
 - Other trap information for this trap identifier
- `apSysMgmtTrapInformationTable`—You can poll this table to obtain further details about the traps recorded in the `apSysMgmtTrapTable` table. The following information appears:
 - Data index
 - Data type
 - Data length
 - The data itself (in octets)

Trap tables do not record information about alarm severity.

The apSysMgmtTrapTable can hold up to 1000 entries, and you can configure the number of days these entries stay in the table for a maximum of seven days. If you set this parameter to 0 days, the feature is disabled. And if you change the setting to 0 days from a greater value, then the Net-Net SBC purges the tables.

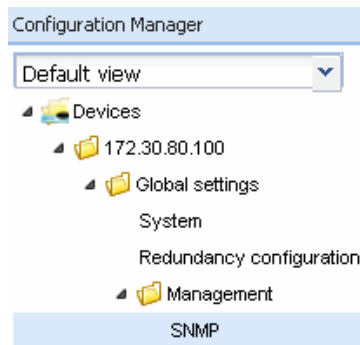
Caveats

Note that the Net-Net SBC does not replicate alarm synchronization table data across HA nodes. That is, each Net-Net SBC in an HA node maintains its own tables.

Configuring Alarm Synchronization

To use alarm synchronization:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click SNMP. The SNMP data window appears in the content area.
3. Scroll to Trap table.
4. **Trap event lifetime (days)**—To enable alarm synchronization—and cause the Net-Net SBC to record trap information in the apSysMgmtTrapTable and the apSysMgmtTrapInformationTable—enter the number of days you want to keep the information. Leaving this parameter set to 0 (default) turns alarm synchronization off. You can keep information in the tables for up to seven days. Seven days is the maximum value for this parameter.

Trap table

Trap event lifetime (days), Enable trap MIB table

5. Click **Apply**.

Refer to the *MIB Reference Guide* for more information.

Configuring TCP Timers

You can configure your Net-Net SBC to detect failed TCP connections more quickly so that data can be transmitted via an alternate connection before timers expire. Across all protocols, you can now control the following for TCP:

- Connection establishment
- Data retransmission
- Timer for idle connections

Configuring TCP Timers

This section explains how to configure each TCP timer, including descriptions of each.

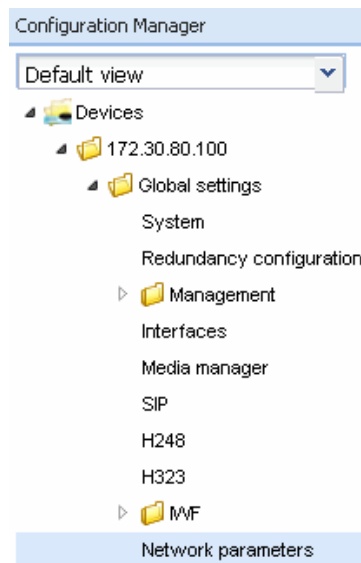
Configuring TCP Connection Establishment

To establish connections, TCP uses a three-way handshake during which two peers exchange TCP SYN messages to request and confirm the active open connection. In attempting this connection, one peer retransmits the SYN messages for a defined period of time if it does not receive acknowledgement from the terminating peer. Now, however, you can configure the amount of time in seconds between the retries as well as how long (in seconds) the peer will keep retransmitting the messages.

You set two options in the network parameters configuration to specify these amounts of time: **atcp-syn-rxmt-interval** and **atcp-syn-rxmt-maxtime**.

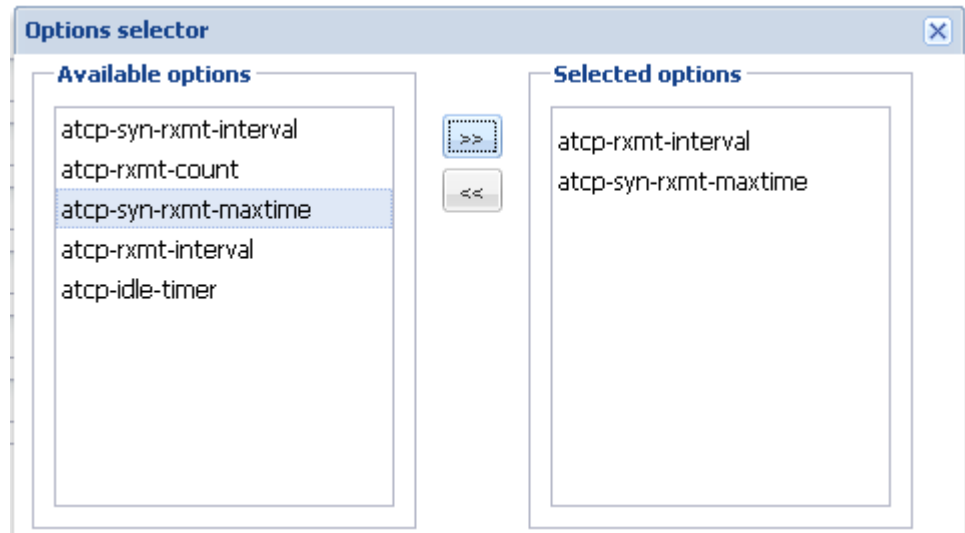
To configure TCP connection establishment:

1. Expand the Configuration Manager slider > Global settings.



2. Click Network parameters.
3. Press **Add** to add the configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. **atcp-syn-rxmt-interval**—Click atcp-syn-rxmt-interval in the Available options list to select it.

6. Click  to move the option to the Selected options list.
7. **atcp-syn-rxmt-maxtime**—Click atcp-syn-rxmt-maxtime in the Available options list to select it.
8. Click  to move the option to the Selected options list.



9. **atcp-syn-rxmt-interval**—Edit the atcp-syn-rxmt-interval textbox to enter the value in seconds you want to use as the interval between TCP SYN messages when the Net-Net SBC is trying to establish a connection with a remote peer. The range is between 2 and 10.
10. **atcp-syn-rxmt-maxtime**—Edit the atcp-syn-rxmt-maxtime textbox to enter the maximum time in seconds for trying to establish a TCP connection. The range is between 5 and 75.
11. Click **OK**.
12. Click **Apply**.

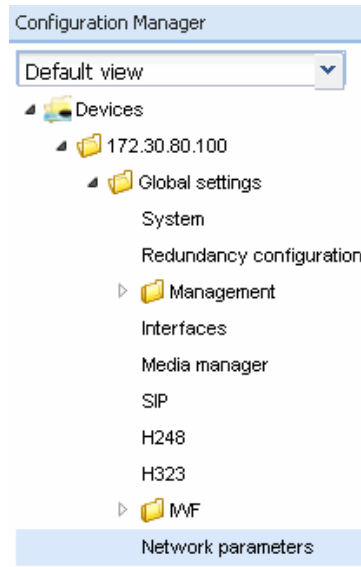
Configuring TCP Data Retransmission

TCP is considered reliable in part because it requires that entities receiving data must acknowledge transmitted segments. If data segments go unacknowledged, then they are retransmitted either until they are finally acknowledged or until the maximum number of retries has been reached. Now you can control both the number of times the Net-Net SBC tries to retransmit unacknowledged segments and the periodic interval (how often) at which retransmissions occur.

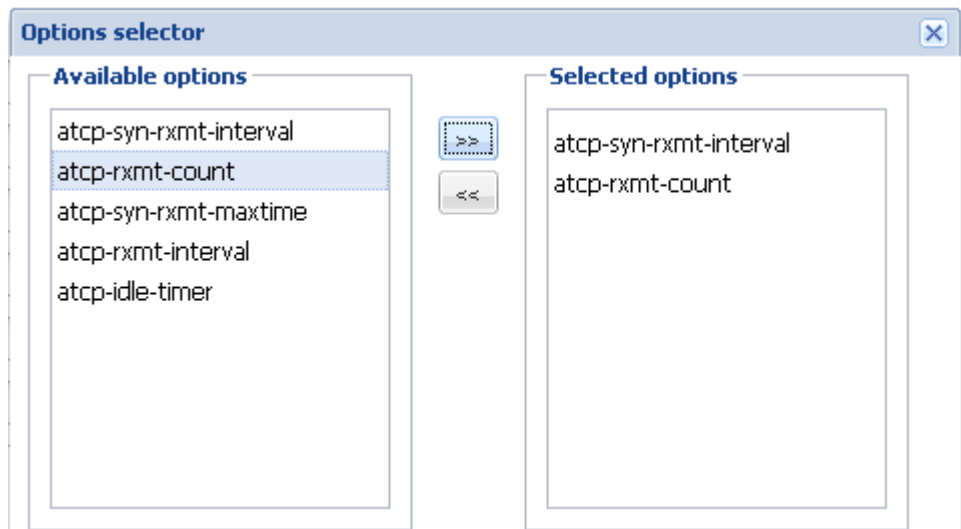
You set two options in the network parameters configuration to specify how many retransmissions are allowed and for how long: **atcp-rxmt-interval** and **atcp-rxmt-count**.

To configure TCP data retransmission:

1. Expand the Configuration Manager slider > Global settings.



2. Click Network parameters.
3. Press **Add** to add the configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. **atcp-syn-rxmt-interval**—Click atcp-syn-rxmt-interval in the Available options list to select it.
6. Click  to move the option to the Selected options list.
7. **atcp-rxmt-count**—Click atcp-syn-rxmt-maxtime in the Available options list to select it.
8. Click  to move the option to the Selected options list.



9. **atcp-rxmt-interval**—Edit the atcp-rxmt-interval textbox to enter the value in seconds you want to use for retransmitting TCP data segments that have not been acknowledged. The range is between 2 and 60.

atcp-rxmt-interval=24

10. **atcp-rxmt-count**—Edit the atcp-rxmt-count textbox to enter the number of times the Net-Net SBC will retransmit a data segment before it declares the connection failed. The range is between 4 and 12.

atcp-rxmt-count=10

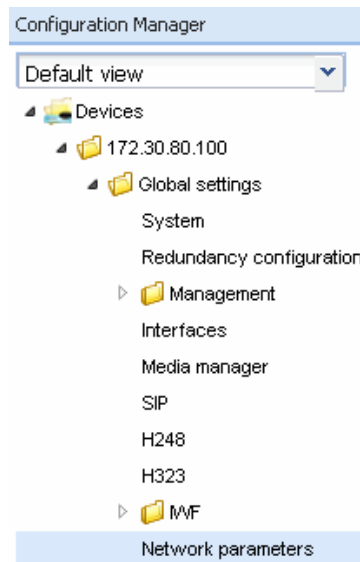
11. Click **OK**.
12. Click **Apply**.

Timer for Idle Connections

When enabled to do so, the Net-Net SBC monitors inbound TCP connections for inactivity. These are inbound connections that the remote peer initiated, meaning that the remote peer sent the first SYN message. You can configure a timer that sets the maximum amount of idle time for a connection before the Net-Net SBC consider the connection inactive. Once the timer expires and the connection is deemed inactive, the Net-Net SBC tries to send a TCP RST message to the remote peer.

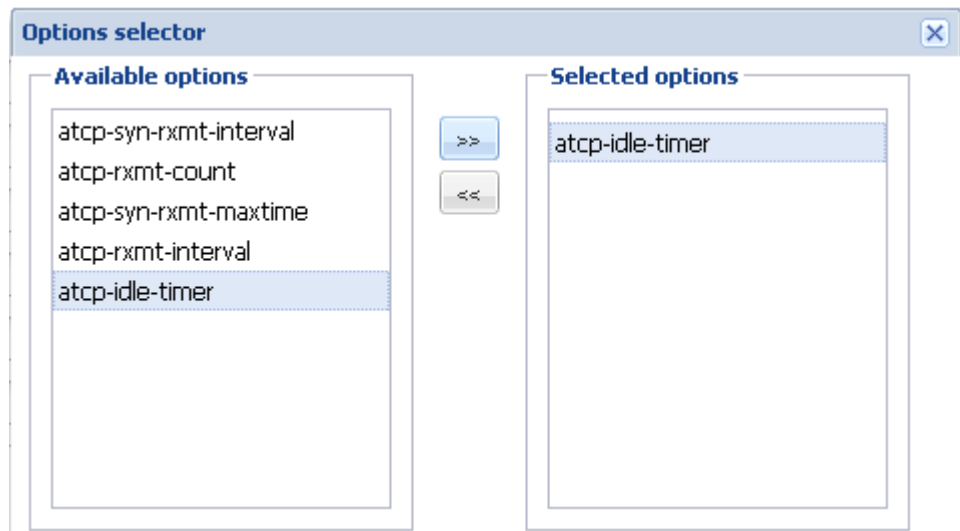
To configure the timer for TCP idle connections:

1. Expand the Configuration Manager slider > Global settings.



2. Click **Network parameters**.
3. Press **Add** to add the configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. **atcp-idle-timer**—Click atcp-idle-timer in the Available options list to select it.

6. Click  to move the option to the Selected options list.



7. **atcp-idle-timer**—Edit the atcp-idle-timer textbox to enter the value in seconds to be used to measure the activity of TCP connections. When the inactivity on a TCP connection reaches this value, the Net-Net SBC declares it inactive. The range is between 2 and 3600.
- 
8. Click **OK**.
9. Click **Apply**.

SIP over SCTP

In releases prior to Release S-C6.1.0, the Net-Net SBC supports UDP and TCP as transport protocols for SIP signaling. Release S-C6.1.0 introduces support for Stream Control Transport Protocol (SCTP). Young in relation to UDP and TCP, SCTP seeks to address some of the shortcomings of the other two transport protocols—most notably by supporting multi-homing and multi-streaming. The Net-Net SBC uses the SIP Via header in the to determine of SCTP should be used as the transport protocol.

For a full description of SCTP, refer to RFC 2960 Stream Control Transmission Protocol.

SCTP Terminology

The following table defines some of the terms commonly found in descriptions of SCTP.

SCTP Term	Definition
SCTP association	Refers to a communication relationship (or logical connection) between SCTP endpoints. SCTP uses a four-way handshake to establish the association between endpoints. This handshake is similar to three-way handshake TCP uses.
Multi-homing	Refers to instances when multiple IP addresses are assigned to a host on the network. Typically, this arrangement entails a host that has multiple network interface cards. (To be supported in future Net-Net OS releases)
Multi-streaming	Refers to the ability to partition data within an association into multiple logical communication channels. Each logical channel—or stream—has the property of independent sequenced delivery. This means that data loss on one stream has no impact on delivery on other streams. (Release S-C6.1.0 supports two incoming streams and two outgoing streams.)
SCTP endpoint	Refers to a logical sender and/or receiver of SCTP packets.

SCTP Overview and Comparisons

A connection-oriented protocol, SCTP uses a four-way handshake to create a connection between two peer entities. This handshake is similar to the three-way handshake TCP uses. SCTP uses the following for messages to establish an association:

1. INIT—Message the client endpoint sends to initiate an association with a peer endpoint
2. INIT-ACK—Message acknowledging the INIT; includes a cookie
3. COOKIE-ECHO—Message that is an echo of the cookie received in the INIT-ACK
4. COOKIE-ACK—Message acknowledging the COOKIE-ECHO

On successful completion of this handshake, the association is established.

How Is SCTP Different from TCP?

While SCTP has many advantages over TCP, the most advertised are multi-streaming and multi-homing.

Multi-streaming allows SCTP to overcome the “head of line” blocking issue that can happen in TCP. In TCP, the loss of a data segment can prevent the delivery of subsequent segments until the lost segment is recovered (this is head of line blocking). SCTP circumvents this problem by supporting multiple associated streams. Data within an association is divided into multiple streams, providing independent and ordered delivery of the data. And so data loss from one stream is prevented from interfering with the data delivery on other streams.

Multi-homing means that an SCTP endpoint can support multiple IP addresses on the same host. In SCTP, application layer software chooses one of the IP address as the primary address. The endpoint should always use the primary address by default, unless specified to do otherwise by the application layer software. If an SCTP endpoint has data to transmit but its primary address is unavailable, the endpoint attempts to transmit its data using one of the alternative IP addresses. This creates a redundancy mechanism transparent to the application layer software.

One minor but notable advantage of SCTP over TCP is the former’s support for message-oriented communication. TCP uses stream-oriented communication that requires the application layer to ascertain message boundaries. SCTP’s message-based communication clearly identifies the beginning and end of data messages.

How Is SCTP Different from UDP?

Both SCTP and TCP are connection-oriented protocols that share some common advantages over UDP: reliable data transfer, congestion control, transport layer fragmentation. Multi-streaming and multi-homing are also SCTP’s advantages over UDP.

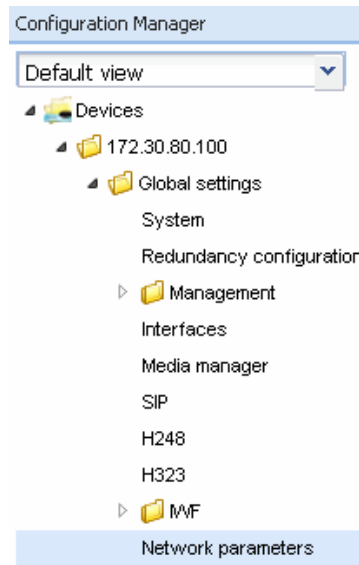
Configuring SCTP Delivery Mode

You can set an SCTP delivery mode to:

- Ordered—Meaning that the endpoint must deliver data according to the stream sequence number
- Unordered—Meaning that the endpoint can deliver data within regard for their stream sequence number

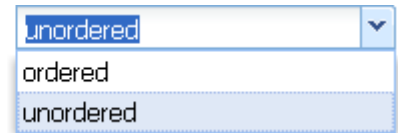
To set the SCTP delivery mode:

1. Expand the Configuration Manager slider > Global settings.



2. Click Network parameters.
3. Press **Add** to add the configuration in the content area.
4. **SCTP send mode**—Click **unordered** (default) to choose it from the combination box so data delivery can occur without regard to stream sequence numbering. If data delivery must follow stream sequence number, click **ordered** from the combination box.

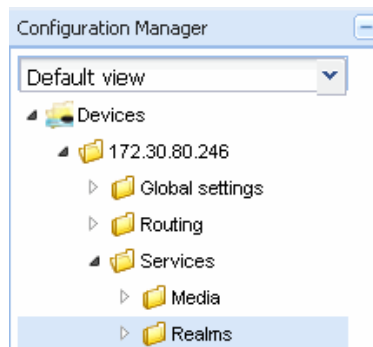
SCTP send mode



5. Click **Apply**.

Configuring SCTP for Realms**To configure SCTP for realms:**

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.

3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to SIP signaling interface table and click **Add**.
The Add SIP signaling interface window appears.
7. **IP address**—Enter the IP address of this SIP signaling interface.
8. **IP port**—Enter the port number of this SIP signaling interface.
 - minimum: 1025
 - maximum: 65535
9. **Transport protocol**—Select SCTP from the combination box..

Add SIP signaling interface

*IP address	0.0.0.0	(Unique required. Enter a valid IP address)
*IP port	5060	(Unique required. Range:1025..65535,..0..0. Default:5060)
*Transport protocol	SCTP	(Unique required. Default:UDP)

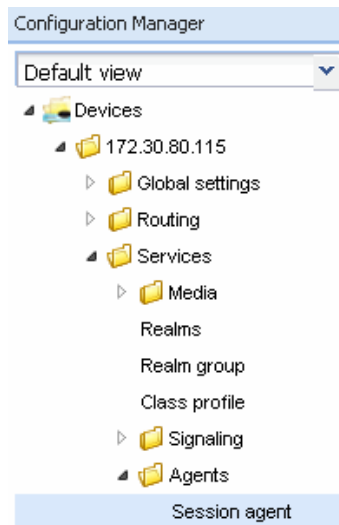
Transport protocol dropdown options: UDP, TLS, SCTP, TCP

10. Click **Apply**.
11. Click **Apply**.

Configuring SCTP for Session Agents

To configure SCTP for session agents:

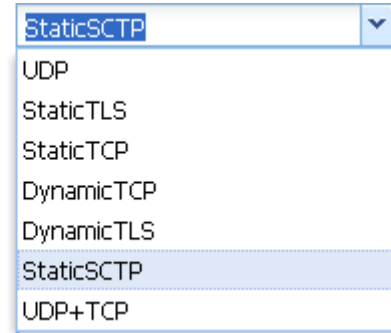
1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a SIP session agent from the table and click **Edit**.

4. **Transport protocol**—Click static SCTP to choose it from the combination box to set static Stream Control Transport Protocol as the transport protocol.

Transport protocol



5. Click **Apply**.

IPv6 Support

IPv6 support has been added to the Net-Net 3800 and Net-Net 4500. Ideally, IPv6 support would be a simple matter of configuring IP addresses of the version type you want in the configurations where you want them. While this is the case for some configuration areas, in others you will need to take care with—for example—the format of your IPv6 address entries or where parameters must be configured with IP addresses of the same version type.

This section explains the changes to the Net-Net Central of which you need to be aware as you start to use IPv6 on your Net-Net 3800 or Net-Net 4500. Note that this first-available implementation of IPv6 is expected to expand in the future; not all configurations and their parameters are available for IPv6 use.

Licensing

IPv6 is a licensed feature on the Net-Net 3800 and Net-Net 4500. If you want to add this license to a system, then contact your Acme Packet sales engineering for information related to the license. Once you have the license information, refer to the *Getting Started* chapter of the *Net-Net Central 4000 Configuration Guide* for instructions on how to add a license.

You do not need to take action if you are working with a new system with which the IPv6 license was purchased.

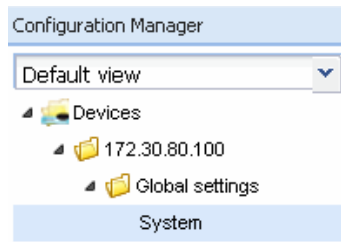
Globally Enabling IPv6

To use IPv6 on your Net-Net 3800 or Net-Net 4500, you need to set the **IPv6 support** parameter in Net-Net SBC system, and then you must reboot your system.

Remember that if you reboot your Net-Net SBC from a Telnet session, you lose IP access and therefore your connection.

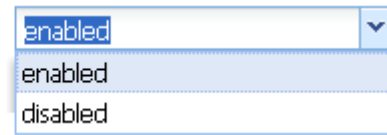
To enable your system for IPv6 support:

1. Expand the Configuration Manager slider > Global settings.



2. Click System. The System window appears in the content area.
3. **IPv6 support**—Select enabled from the combination box to **enable** use for IPv6 on your system. Otherwise, you can leave this parameter set to **disabled** (default).

IPv6 support



4. Click **Apply**.
5. Reboot your Net-Net 3800/ Net-Net 4500 for changes to take effect.

IPv6 Address Configuration

This section calls out the configurations and parameters for which you can enter IPv6 addresses. In this first IPv6 implementation, the complete range of system configurations and their parameters are available for IPv6 use.

The Net-Net SBC follows RFC 3513 its definition of IPv6 address representations. Quoting from that RFC, these are the two forms supported:

- The preferred form is x:x:x:x:x:x:x, where the 'x's are the hexadecimal values of the eight 16-bit pieces of the address. Examples:

FEDC:BA98:7654:3210:FEDC:BA98:7654:3210

1080:0:0:0:8:800:200C:417A

It is not necessary to write the leading zeros in an individual field, but there must be at least one numeral in every field (except for the case described in 2.).

- Due to some methods of allocating certain styles of IPv6 addresses, it will be common for addresses to contain long strings of zero bits. In order to make writing addresses containing zero bits easier, a special syntax is available to compress the zeros. The use of "::" indicates one or more groups of 16 bits of zeros. The "::" can only appear once in an address. The "::" can also be used to compress leading or trailing zeros in an address. For example, the following addresses: 1080:0:0:0:8:800:200C:417A a unicast address FF01:0:0:0:0:0:0:101 a multicast address

0:0:0:0:0:0:0:1 the loopback address

0:0:0:0:0:0:0:0 the unspecified addresses

may be represented as:

1080::8:800:200C:417A a unicast address
 FF01::101 a multicast address
 ::1 the loopback address
 :: the unspecified addresses

Access Control These are the IPv6-enabled parameters in **Access control**.

Parameter	Entry Format
Exception source	<ip-address>[/<num-bits>][:<port>[/<port-bits>]]
Exception destination	<ip-address>[/<num-bits>][:<port>[/<port-bits>]]

Host Route These are the IPv6-enabled parameters in **Host routes**.

Parameter	Entry Format
IP address	<ipv4> <ipv6>
Subnet mask	<ipv4> <ipv6>
Gateway IP address to destination network	<ipv4> <ipv6>

Local Policy These are the IPv6-enabled parameters in **Local policy**.

Parameter	Entry Format
From address	<ipv4> <ipv6> POTS Number, E.164 Number, hostname, wildcard
To address	<ipv4> <ipv6> POTS Number, E.164 Number, hostname, wildcard

Interfaces These are the IPv6-enabled parameters in **Interfaces**.

Parameter	Entry Format
Host name	<ipv4> <ipv6> hostname
IP address	<ipv4> <ipv6>
HA utility IP address #1	<ipv4> <ipv6>
HA utility IP address #2	<ipv4> <ipv6>
Subnet mask	<ipv4> <ipv6>
Primary IP address	<ipv4> <ipv6>
Secondary IP address	<ipv4> <ipv6>
Primary	<ipv4> <ipv6>
First backup	<ipv4> <ipv6>
Second backup	<ipv4> <ipv6>

Parameter	Entry Format
HIP IP addresses	<ipv4> <ipv6>
ICMP addresses	<ipv4> <ipv6>

Realms These are the IPv6-enabled parameters in **Realms**.

Parameter	Entry Format
IP address prefix	[<ipv4> <ipv6>]/prefix

Session Agent These are the IPv6-enabled parameters in **Session agents**.

Parameter	Entry Format
Name(FQDN)	<ipv4> <ipv6>
IP address (required for SIP pings to test SA availability)	<ipv4> <ipv6>

SIP These are the IPv6-enabled parameters in the SIP configuration.

Parameter	Entry Format
Host name/IP address	<ipv4> <ipv6> hostname *

SIP Service>SIP Signaling Interface These are the IPv6-enabled parameters in the SIP service>SIP signaling interface configuration.

Parameter	Entry Format
IP address	<ipv4> <ipv6>

Realm Media Address These are the IPv6-enabled parameters in the **Realm media address** configuration.

Parameter	Entry Format
IP address	<ipv4> <ipv6>

SBC System These are the IPv6-enabled parameters in **SBC System**.

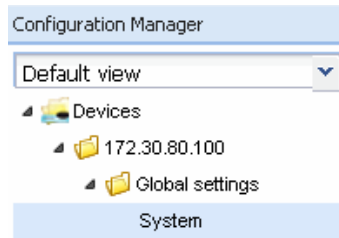
Parameter	Entry Format
Default IPv6 gateway	<ipv6>

IPv6 Default Gateway

You configure a default gateway with a parameter that now has its own IPv6 equivalent.

To configure an IPv6 default gateway:

1. Expand the Configuration Manager slider > Global settings.



2. Click System. The System window appears in the content area.
3. **Default IPv6 gateway**—Enter the IPv6 default gateway for this Net-Net SBC. This is the IPv6 egress gateway for traffic without an explicit destination.

Default IPv6 gateway

4. Click **Apply**.

Network Interfaces and IPv6

You set many IP addresses in the network interface, one of which is the specific IP address for that network interface and others that are related to different types of management traffic. This section outlines rules you must follow for these entries.

- **Interface, IP address** parameter, set a single IP address. When you are working with an IPv6-enabled system, however, note that all other addresses related to that network-interface IP address must be of the same version.
- Heterogeneous address family configuration is prevented for the **Primary**, **First backup**, and **Second backup** parameters.
- HIP addresses, use either IPv4 or IPv6 entries.
- ICMP addresses, use either IPv4 or IPv6 entries.
- Telnet, FTP, and SNMP, do not use IPv6; your entries **MUST** use IPv4.

Access Control List Support

The Net-Net SBC supports IPv6 for access control lists in two ways:

- For static access control lists that you configure in the **Access control** configuration, your entries can follow IPv6 form. Further, this configuration supports a prefix that enables wildcarding the source IP address.
- Dynamic ACLs are also supported; the Net-Net SBC will create ACLs for offending IPv6 endpoints.

Data Entry

When you set the **Exception source** and **Exception destination** parameters in the **Access control** configuration, you will use a slightly different format for IPv6 than for IPv4.

For the **Exception source**, your IPv4 entry takes the following format: `<ip-address>[/<num-bits>][:<port>[/<port-bits>]]`. And for the **Exception source**, your IPv4 entry takes this format: `<ip-address>[:<port>[/<port-bits>]]`.

Since the colon (:) in the IPv4 format leads to ambiguity in IPv6, your IPv6 entries for these settings must have the address encased in brackets ([]):

`[7777::11]/64:5000/14.`

In addition, IPv6 entries are allowed up to 128 bits for their prefix lengths.

DNS Support

The Net-Net SBC supports the DNS resolution of IPv6 addresses; in other words, it can request the AAAA record type (per RFC 1886) in DNS requests. In addition, the Net-Net SBC can make DNS requests over IPv6 transport so that it can operate in networks that host IPv6 DNS servers.

For mixed IPv4-IPv6 networks, the Net-Net SBC follows these rules:

- If the realm associated with the name resolution is an IPv6 realm, the Net-Net SBC will send the query out using the AAAA record type.
- If the realm associated with the name resolution is an IPv4 realm, the Net-Net SBC will send the query out using the A record type.

In addition, heterogeneous address family configuration is prevented for the **Primary**, **First backup**, and **Second backup** parameters.

Homogeneous Realms

IPv6 is supported for realms and for nested realms, as long as the parent chain remains within the same address family. If you try to configure realms with mixed IPv4-IPv6 addressing, your system will issue an error message when you try to save your configuration.

Compact Flash Memory Use With Local CDRs / FTP Push

The Net-Net 9200 can use a compact flash memory card that extends the system's internal storage beyond the fixed amount of flash RAM. When configuring local CDR creation, you can configure the Net-Net SBC to use compact flash memory for local CDR file storage instead of the limited internal flash RAM.

Disk space on the compact flash module appears as a local volume on the Net-Net SBC. Wherever you specify a volume name for a configuration parameter value, you can enter a volume located on the compact flash module, (unless the parameter is otherwise specified).

Compact Flash Memory Specifications

The Net-Net 9200 copies CDR files from RAM drive to a compact flash memory module for persistent CDR storage. Compact Flash memory is inserted into the appropriately labeled slot on the Net-Net 9200's MIU.

Only one PCMCIA Compact Flash device is supported: the SanDisk SDCFX3-016G-A31 16 GB Extreme III CompactFlash Card.

Flash devices must be formatted FAT32, which can be performed by qualified Acme Packet personnel.

Compact Flash Standby Replication

Because PCMCIA is permanent storage, CDR files in PCMCIA are not replicated to standby. In addition, CDR files on the active SPU/MIU pair's internal flash drive are not replicated to standby. To force the Net-Net 9200 to transfer files from the standby's FLASH, it is necessary to transition the standby card to the ACTIVE role.

Local CDR Storage Directory

To save local CDR files to the compact flash module, configure the **file path** parameter in the account config with **/pcmcia**. This indicates that the files should be written to the compact flash module which attaches to the PCMCIA bus.

FTP Push Backup

When FTP push is enabled, if all FTP push servers are unreachable, then local CDR files are written to local file system until the FTP push servers return to service. Once an FTP Push server becomes reachable, the Net-Net SBC transfers all local CDR files to the remote server automatically. After a local CDR file has been successfully transferred to the FTP server from the Net-Net SBC, it is deleted from the local volume.

Local CDR File Compression

You can configure the Net-Net SBC to compress local CDRs in .gz format to save disk space. The local CDRs will be compressed and appear with a .gz file extension. This feature is enabled with the **file compression** parameter.

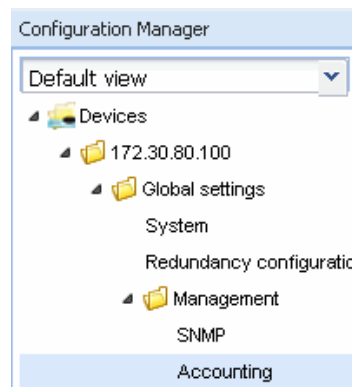
CDR Decompression

The Net-Net 9200 can decompress locally compressed .gz files before sending them to a remote file server. Only files with a ".gz" extension will be decompressed. Files lacking this extension will be transmitted as-is. If an error occurs during decompression, the file will be transmitted compressed (with the .gz extension). Setting the file-decompression parameter to **enabled** will push decompressed CDR files to the remote server.

Writing Local CDRs to a Compact Flash Module

To configure a Net-Net 9200 to write local CDRs to a compact flash module:

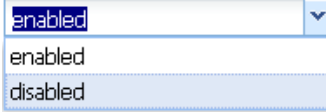
1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Click the Push and file based accounting tab.

5. **File compression**—Click **enabled** in the drop down list to compress local CDR files in .gz format. The default value is disabled.

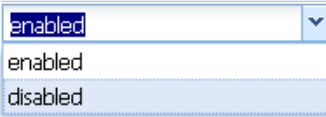
File compression



The screenshot shows a web form with a label 'File compression' and a dropdown menu. The dropdown menu is open, showing two options: 'enabled' (which is highlighted) and 'disabled'.

6. **File decompression**—Click **enabled** in the drop down list for the Net-Net 9200 to decompress local CDR files in memory before pushing them the off box server. The default value is disable.

File decompression



The screenshot shows a web form with a label 'File decompression' and a dropdown menu. The dropdown menu is open, showing two options: 'enabled' (which is highlighted) and 'disabled'.

7. Click **Apply**.

Compact Flash Module Monitoring

Low Disk Space Warning

The Net-Net SBC can initiate an alarm and an SNMP trap when a volume reaches a configured threshold of remaining free disk space, configured as a percentage of volume. You can configure multiple alarms, each with increasing severity that indicate less free disk space.

Low Disk Space Threshold Alarm

The low disk space threshold alarm is configured in **alarm threshold** configuration element. It is non-health affecting. The threshold alarm appears as follows:

```
SYSTEM# show alarm current
1 alarms to show
ID      Task      Severity      First Occurred      Last Occurred
131142  547896076      4      2009-08-25 13:36:26  2009-08-25 13:36:26
Count   Description
1       Volume /pcmcia space used 81% is over major threshold of 80%.
```

Low Disk Space Threshold SNMP Trap

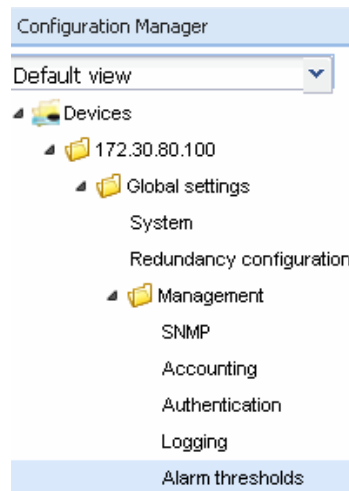
For any threshold reached, an SNMP trap will be sent to all configured trap-receivers. The apSysMgmtStorageSpaceAvailThresholdTrap trap contains the following information:

- VolumeName—name of the volume where a threshold was exceeded
- CurrentValue—current percentage of disk space value that is exceeding one of the thresholds.
- MinorThreshold—configured minor threshold for this volume, if none then this is 0.
- MajorThreshold—configured major threshold for this volume, if none then this is 0.
- CriticalThreshold—configured critical threshold for this volume, if none then this is 0.

Configuring Alarm Thresholds for Monitoring Free Disk Space

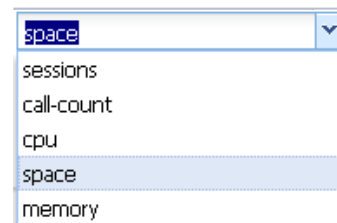
To configure alarm thresholds for monitoring free disk space:

1. Expand the Configuration Manager slider > Global settings > Management.



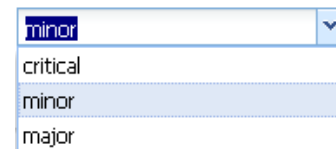
2. Click Alarm thresholds. The Alarm threshold table appears in the content area.
3. Click **Add**. The Add-alarm threshold dialog box appears.
4. **Alarm type**—Click **space** in the drop down list to create an alarm based on reduced free disk space.

*Alarm type



5. **Severity**—Enter the severity level of this alarm. Valid severity values are MINOR, MAJOR, and CRITICAL. The default value is MINOR.

*Severity



6. **Volume**—Enter volume name you wish to monitor. Generally this string should be the same as the **file-path** parameter located in the account-config.

*Volume

7. Click **Apply**.

Multiple CDR Push Receivers

Your Net-Net SBC supports up to five CDR push receivers for use with local file storage and FTP push (although Net-Net EMS allows for more than five). For each receiver you configure, you can set the file transfer protocol you want to use—either FTP or SFTP. The system uses the push receivers according to the priorities you set by giving a 0 through 4 priority number to the server when you configure it; 0 is the highest priority, and 4 is the lowest. By default, push receivers always have their priority at the lowest setting (4).

Based on the priority level you set, the Net-Net SBC uses a strategy (which you also set) to select a CDR push receiver. If the highest priority push receiver selected using the strategy becomes unavailable, for example, times out, the Net-Net SBC uses the strategy (hunt, round robin, and so on) to select another.

When you change the configuration, the Net-Net SBC updates the list of push receivers if it has changed. If you decide not to configure CDR push receivers but enable FTP push anyway, the system pushes CDRs to the location the **FTP address** and **ftp-port** parameters define.

The Net-Net SBC sends traps when a single push receiver or all push receivers become unavailable.

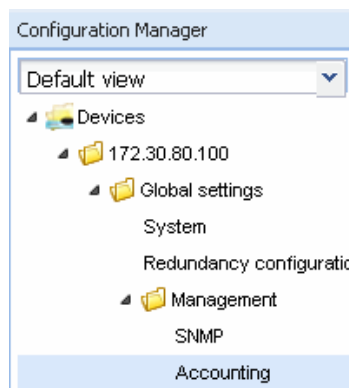
- When one CDR push receiver becomes unavailable, the **CDR_PUSH_RECEIVER_FAIL_TRAP** trap is sent and a minor alarm is generated.
- When all of the configured CDR push receivers become unavailable, the **CDR_ALL_PUSH_RECEIVERS_FAIL_TRAP** is sent and a major alarm is generated.

When one or more of the push receivers comes back, the **CDR_ALL_PUSH_RECEIVERS_FAIL_CLEAR_TRAP** is sent and the alarm is cleared.

Configuring CDR Push Receivers

To configure multiple CDR push receivers:

1. Expand the Configuration Manager slider > Global settings > Management.



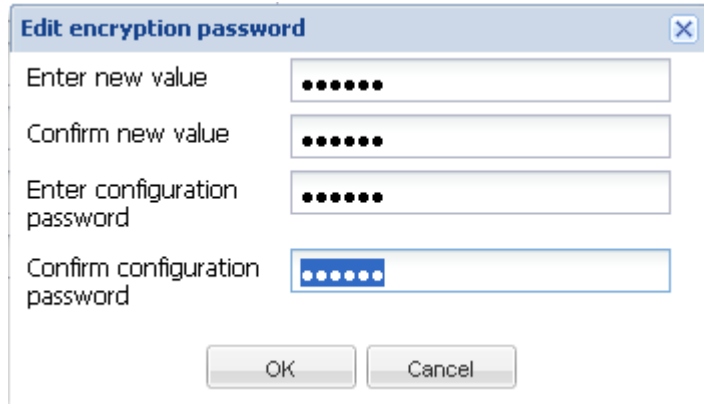
2. Click Accounting.

3. Press **Add** to add the accounting configuration in the content area.
4. Click the Push and file based accounting tab at the top of the content area.
5. Scroll to the CDR push servers table and click Add. The Add push-receiver dialog box appears.
6. **Server address**—Enter the IP address of the FTP/SFTP server to which you want the Net-Net SBC to push CDR files.

Note: The Net-Net SBC allows up to five push receivers to be configured.

7. Click **Apply**. The CDR push server is added to the CDR push server table.
8. Select the CDR push server in the table and click **Edit**.
9. **Admin state**—Click the check box to enable this FTP/SFTP push receiver if you want the Net-Net SBC to send CDR files to it. By default, this parameter is set to **enabled**.
10. **Server port**—Enter the port number on the FTP/SFTP server for which you set the IP address in Step 5. The Net-Net SBC will send CDR files to this port on the designated FTP/SFTP server. There is no default for this parameter.
11. **Remote pathname**—Enter the remote pathname to which you want CDR files to be sent on the push receiver. CDR files will be placed in this location on the FTP/SFTP server. There is no default for this parameter.
12. **Remote filename prefix**—Enter the filename prefix (as a string) to prepend it to the CDR files the Net-Net SBC sends to the push receiver. The Net-Net SBC does not rename local files. There is no default for this parameter.
13. **Transfer protocol**—Enter SFTP if you want to change the transport protocol for this push receiver from its default, **FTP**.
14. **User name**—Enter the username the Net-Net SBC is to use when connecting to this push receiver. There is no default for this parameter.
15. **Password**—Click **Edit**. The Edit password dialog box appears.
16. **Enter new value**—Enter the password corresponding to the user name you configured in Step 13 so the Net-Net SBC can access the push receiver. There is no default for this parameter. The password echoes in the text box.
17. **Confirm new value**—Retype the password corresponding to the user name you configured in Step 13 to confirm it. The password echoes in the text box.
18. **Configuration password**—Enter the password used to encrypt the configuration information on the Net-Net SBC.
19. **Confirm configuration password**—Retype the configuration password to confirm the password.

20. Click **OK**. A message appears recommending you run the ACLI command **verify-config** on the Net-Net SBC after you save your configuration



The dialog box titled "Edit encryption password" contains four input fields, each with a masked password of seven asterisks. The fields are labeled: "Enter new value", "Confirm new value", "Enter configuration password", and "Confirm configuration password". At the bottom are "OK" and "Cancel" buttons.

21. Click **OK** to clear the message. The asterisks that represent the password appear in the field next to password..
22. **SFTP public key**—Click the SFTP public key profile to use for authentication in the combination box when you have defined this as an SFTP push receiver. Otherwise, the Net-Net SBC will use password authentication.
23. **Receiver priority**—Enter 0 through 4 to set the priority of this push receiver in relation to the others you configure on the system. Because you can configure up to five (5) push receivers, you define a push receiver's priority by giving it a number. The highest priority—and the push receiver the system will use first—is 0. The lowest priority—and the push receiver the system will use last—is 4. This parameter defaults to 4.
24. **Temporary remote file**—Enable to upload CDR CSV files initially with a tmp-prefix and then rename the file after successful upload.

Admin state	enabled
*Server address	100.10.20.30
Server port	21
Remote pathname	
Remote filename prefix	
Transfer protocol	sftp
User name	
Password	
SFTP public key	
Receiver priority	4
Temporary remote file	disabled

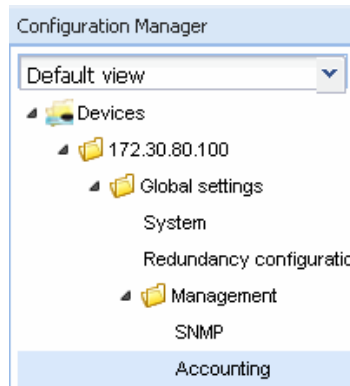
25. Click **Apply**.

Configuring a Push Receiver Fallback Method

You set the push receiver strategy and define the maximum timeout in seconds in the accounting tab.

To configure a push receiver fallback method:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Click the Push and file based accounting tab at the top of the content area.
5. **Multiple push receiver usage strategy**—Set the strategy you want the Net-Net SBC to use when selecting from multiple push receivers. The default is **hunt**.

Strategy	Description
Hunt	The Net-Net SBC selects the push receiver from the available list according to the priority level. The system uses this strategy as its default.
Failover	The Net-Net SBC selects the push receiver based on priority level and will continue to use that same push receiver until it fails over.
RoundRobin	The Net-Net SBC selects push receivers systematically one after another, balancing the load among all responsive push receivers.
FastestRTT	The Net-Net SBC selects the push receiver based on best average throughput. For this situation, throughput is the number of bytes transferred divided by the response time. The system uses a running average of the five most recent throughput values to accommodate for network load fluctuations.

6. **Maximum wait time for timeout for push receivers (sec)**—Enter the amount of time in seconds to wait before the Net-Net SBC declares a push receiver to have failed over. This default value for this parameter is **60**.

Multiple push receiver usage strategy	Failover
Maximum wait time for timeout for push receivers (sec)	120

7. Click **Apply**.

RADIUS Account Server Prioritization

Especially useful for customers who use multiple Net-Net SBCs, the RADIUS account server prioritization feature allows you to assign a priority to each of the account servers you configure. Setting the priority for RADIUS accounting servers allows you to load balance traffic across the servers.

Without this feature, the Net-Net SBC sorts RADIUS accounting servers by their IP addresses and ports. For example, if you have a pre-existing accounting server with the IP address and port combination of 10.1.31.2:1813 and then configure a new server at 10.0.3.12:2145, the new server will take priority over the pre-existing one. Of course, you always have the option of allowing the system to set the priority or your accounting servers in this way.

The prioritization feature works with all of the strategy types you set in the accounting configuration. However, it is most applicable to the **hunt** or **failover** strategies. You can assign a number to each server to mark its priority, or you can leave the priority parameter set to 0 (default) so the Net-Net SBC prioritizes them by IP address and port.

Server Prioritization Example

This example shows you how you might benefit from using the prioritization feature if you have multiple Net-Net SBCs sending RADIUS CDRs to multiple RADIUS servers. Consider the following Net-Net SBCs and accounting servers.

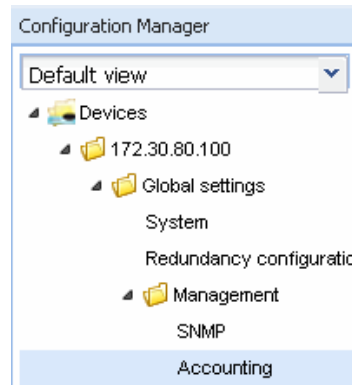
Net-Net SBC	Account Server1 Priority	Account Server2 Priority	Account Server3 Priority
Net-Net SBC1	10	7	4
Net-Net SBC2	7	4	10
Net-Net SBC3	4	10	7
Net-Net SBC4	10	7	4
Net-Net SBC5	7	4	10
Net-Net SBC6	4	10	7

If the strategy for this example is set to **hunt** or **failover** and assuming no timeouts are pending, you can see that Net-Net SBC1 sends its accounting traffic to Account Server3 over the other two. Net-Net SBC2 sends its traffic to Account Server2 over the others, and likewise for the remainder of Net-Net SBCs and servers. The traffic, then, is load balanced across the servers, less likely to overburden any of them.

Setting Priority for Account Servers

To set the priority for account servers:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Click the Radius accounting tab at the top of the content area.
5. Select an accounting server name in the table and click **Edit**. The Edit Accounting server window appears.
6. **Priority**—Enter the number corresponding to the priority you want this account server to have in relation to the other account servers to. The default for this parameter is 0, meaning the prioritization feature is turned off—the Net-Net SBC will prioritize accounting servers by IP address and port. If you do not retain the default, you can use any numbering scheme that suits your needs and ease of use.

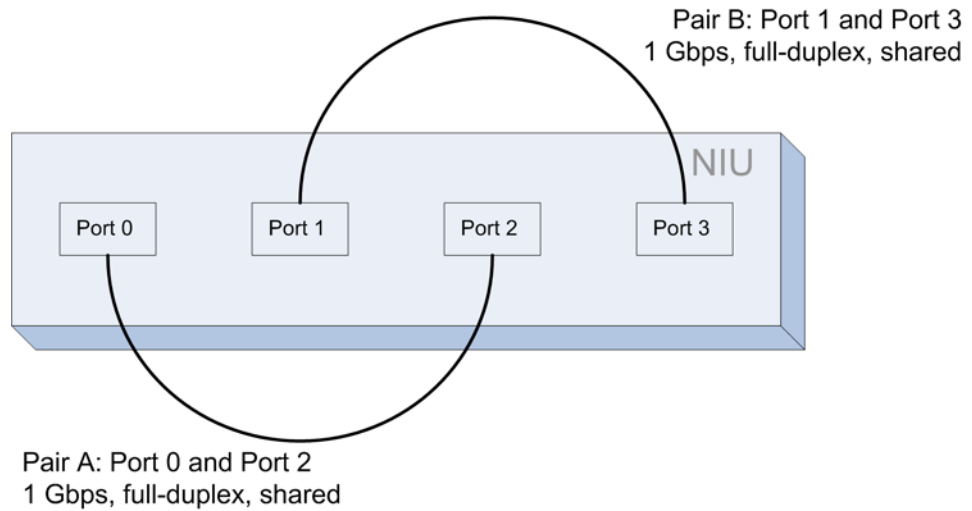
Priority

7. Click **Apply**.

NIU GigE Utilization

The Net-Net 9200's NIU physical interfaces operate in a Shared-1000Mbps per-port-pair mode. This enables each pair of physical ports on an NIU to share 1Gbps of full-duplex bandwidth. If only one of the pair's ports is used, then it is guaranteed 1 Gbps of full-duplex bandwidth. When both ports in a port pair are in-use, a single port can use more than half of the total bandwidth as long as the other port is using less than the remaining unused part of the total bandwidth (1Gbps). For example: If both ports comprising a port pair are in use, one port can burst to over 500 Mbps if its sibling pair is operating at below the 500 Mbps threshold.

The following illustration shows the matched port pairs: Port 0 and Port 2 comprise port pair A; Port 1 and Port 3 comprise port pair B.

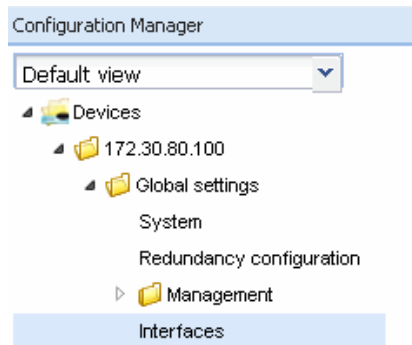


To ensure that a port has a full 1 Gbps bandwidth available to itself, disconnect (or disable) its sibling port within the port pair.

Segmenting Traffic

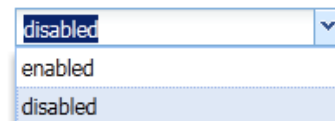
In situations where a port pair should be segmented between signaling and media traffic, use the following configuration technique on the network interface port dedicated to signaling traffic:

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.
3. Select the physical interface you would like to use and click **Edit**.
4. **Auto-negotiation**—Select disabled from the combination box.

Auto-negotiation - 10/100Mbps



5. **Speed**—Select 100Mbps from the combination box.

Bandwidth (Mbps)

100	▼
10	
100	
1000	

This technique leaves 900Mbps dedicated for media traffic.

6. Click **Apply**.

SNMPv3 Secure Traps

The Net-Net SBC supports SNMPv3, which provides the SNMP agent and SNMP Network Management System (NMS) with authentication, privacy, and access control during the delivery of secured traps. Currently, SNMPv3 traps are supported on the Net-Net SBC; SNMPv3 Get/Get-Bulk/Set actions are not supported at this time.

By default, the Net-Net SBC supports SNMPv1v2. If you want to retain existing SNMPv1v2 behavior, you do not need to update configuration. You can enable SNMPv3 at any time, at which point SNMPv1v2 configurations are ignored, and only SNMPv3 encrypted traps are sent to associated external SNMP managers. **Snmp-agent-mode**, an attribute under **system-config**, allows you to select the desired mode.

Authentication and Privacy

SNMPv3 employs a User-Based Security Model (USM). The two protocols used for authentication and privacy are:

- Authentication—HMAC-SHA-96
- Privacy—CBC-DES

Four parameters generate keys under the designated algorithm:

- **SNMPEngineID**—The unique identifier for the SNMP Engine. This value is a specially formatted string for use in the SNMP.
- **User name**—The user's name as defined under **snmp-user-entry**.
- **Authorization password**—The authorization password configured under the **snmp-user-entry** configuration. This parameter is used to derive the authentication key.
- **Privacy password**—You set the privacy password in the **snmp-user-entry** configuration. It is used to derive the password key.

Password-to-Key Conversion

There are two distinct passwords in SNMPv3. The authentication password is manipulated using the HMAC-SHA-96 algorithm to produce a key used to authenticate the trap. Authentication ensures the identity of the user and that the trap has not been tampered with in transit. Likewise, the privacy password is manipulated using the CBC-DES algorithm to ensure message privacy.

One user is associated by a name, an authentication password and a privacy password. These three parameters are always consistent for the user and can be used across multiple SBCs. The key generation differs from one SBC to another due to the

varying SNMPEngineIDs. This ensures that a compromised key for one SBC does not compromise the keys for other SBCs associated with the same user.

Enabling SNMPv3

The table below gives a brief overview of the SNMPv3 configuration on your Net-Net SBC. The Caveats column describes the SNMPv1v2 configuration attributes that are ignored if **SECURE-TRAP** mode is enabled.

Configuration	Description	Caveat
<code>snmp-agent-mode</code>	Set this attribute to SECURE-TRAP to enable SNMPv3.	Once SNMPv3 is enabled, the snmp-community and community-name attributes are ignored.
<code>snmp-engine-id-suffix</code>	Set this attribute as a string to customize and uniquely identify the SNMP Engine.	The show snmp-info command has been expanded to include the SNMP Engine Base, the SNMP Engine Suffix, and the SNMP Engine ID.
<code>snmp-user-entry</code>	Enter the user name, authorization password and privacy password.	The user, as defined in this object, must be added to the attribute user-list under trap-receiver in order to receive secured traps.
<code>trap-receiver</code>	Configure a trap-receiver with the IP address of the NMS that receives secured traps.	
<code>user-list</code>	Add users who are authorized to receive secured traps.	If instances of snmp-user-entry are configured, but no users are listed under user-list , a warning message is sent during a verify-config execution.

Retaining Existing SNMPv1v2 Behavior

If you are upgrading to software version C6.3.0 or above and want to retain your existing SNMP configurations, you do not need to take any action. The Net-Net SBC sets **snmp-agent-mode** to **V1V2** by default, disabling all SNMPv3 configurations.

Downgrading Software After Enabling SNMPv3

If you enable SNMPv3 on an SBC running C6.3.0 or above, and you downgrade to a previous software version, the software does not recognize the SNMPv3 configuration objects or attributes.

Consideration for HA Nodes

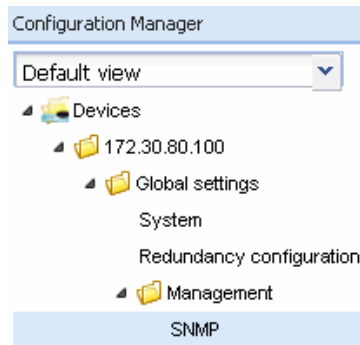
Key pairs are generated based on the user and SNMPEngineID. In the event of a switchover, the SNMPEngineID will vary. The user's NMS should be updated with the SNMPEngineID of the standby SBC. No action is required on the SBC.

Enabling SNMPv3

This section shows you how to enable SNMPv3 on your system, how to add users, and how to add users to authorized trap receivers.

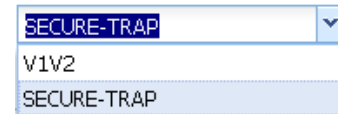
To enable SNMPv3 on the Net-Net SBC for sending secured traps:

1. Expand the Configuration Manager slider > Global settings > Management.



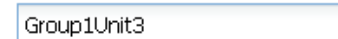
2. Click SNMP. The SNMP data window appears in the content area.
3. **Agent mode**—Click **SECURE-TRAP** in the drop-down list to enable support. The default value is V1V2.

Agent mode



4. **Engine id suffix**—To set a unique suffix for the SNMP EngineID, enter a string. This attribute is optional.

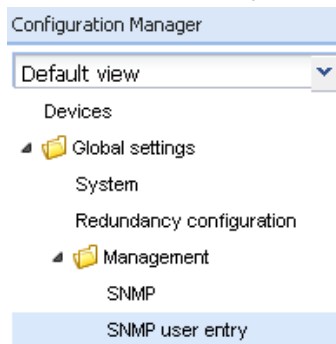
Engine id suffix



5. Click **Apply**.

Users and Password Configuration**To configure users for SNMPv3:**

1. Expand the Configuration Manager slider > Global settings > Management.
2. Click SNMP user entry.



3. Click **Add**. The Add SNMP user entry dialog box appears.
4. **User name**—Enter the name for this user. This value is required and must be unique.
5. **Privacy password**—Click **Edit**. The Edit encryption password dialog box appears.
 - 5a. Enter a new value. This value is required.

- 5b. Confirm new value.
- 5c. Enter configuration password. This value is required.
- 5d. Confirm configuration password.

Edit encryption password [X]

Enter new value

Confirm new value

Enter configuration password

Confirm configuration password

OK Cancel

- 6. Click OK.
- 7. The following Info dialog box appears:

Info

 It is recommended that you run verify-config on SBC after saving this config to the SBC.

OK

- 8. Click OK.
- 9. **Authentication password**—Click Edit. The Edit encryption password dialog box appears.
 - 9a. Enter a new value. This value is required.
 - 9b. Confirm new value.
 - 9c. Enter configuration password. This value is required.
 - 9d. Confirm configuration password.

Edit encryption password [X]

Enter new value

Confirm new value

Enter configuration password

Confirm configuration password

OK Cancel

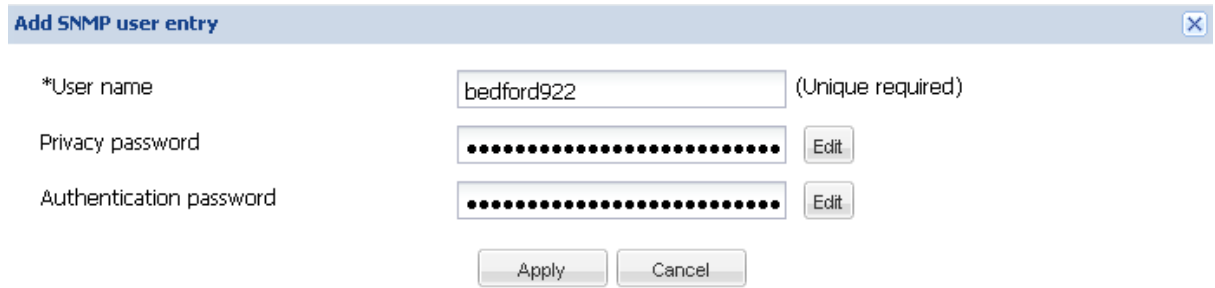
- 10. The following Info dialog box appears:

Info

 It is recommended that you run verify-config on SBC after saving this config to the SBC.

OK

- Click **OK**.



The dialog box titled "Add SNMP user entry" contains the following fields and controls:

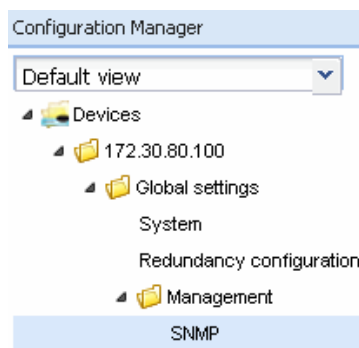
- *User name:** A text field containing "bedford922" with a "(Unique required)" label to its right.
- Privacy password:** A password field with masked characters and an "Edit" button to its right.
- Authentication password:** A password field with masked characters and an "Edit" button to its right.
- At the bottom, there are "Apply" and "Cancel" buttons.

- Click **Apply**.

Adding Authorized Trap Receivers

To add users as authorized trap-receivers:

- Expand the Configuration Manager slider > Global settings > Management.



- Click **SNMP**. The SNMP data window appears in the content area.
- Scroll to the Trap receiver table.
- Click **Add**. The Add trap receiver dialog box appears.
- IP address:port**—Enter the IP address and port for the NMS that supports SNMPv3.
- Community name**—Enter a community name for this trap receiver.
- User list**—Click (+) or (-) in the drop-down list to add or subtract users to the list using (+) and (-) symbols.
- Click **Apply**.

Protection Against DDoS Attack From Behind a NAT

Configuring Protection Against DDoS Attacks From Behind a NAT

You configure this parameter in the Access control, Realms and Media manager.

Access Control

To configure protection against DDoS attacks from behind a NAT in Access control:

1. Expand the Configuration Manager slider > Services > Signaling.



2. Click Access control.
3. Click Add.
4. Select the Access control configuration you want to configure and click **Edit**.
5. **Maximum number of endpoints behind a NAT device**—Enter the maximum number of endpoints behind a NAT device. The valid range is 0–65535. The default value is 0.

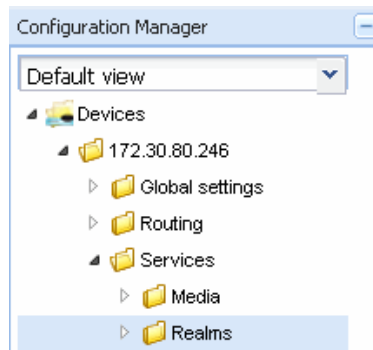
Maximum number of endpoints behind a NAT device

6. Click **Apply**.

Realms

To configure protection against DDoS attacks from behind a NAT in Realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure and click **Edit**.
4. **Maximum number of endpoints behind a NAT device**—Enter the maximum number of endpoints behind a NAT device. The valid range is 0-65535. The default value is 0.

Maximum number of endpoints behind a NAT device

5. **Acceptable number of invalid messages from behind a NAT**—Enter the acceptable number of invalid messages from behind a NAT device. The valid range is 0-65535. The default value is 0.

Acceptable number of invalid messages from behind a NAT

6. **Wait time for invalid register**—Enter the wait time for an invalid register. The valid range is 0, 4-300. The default value is 0.

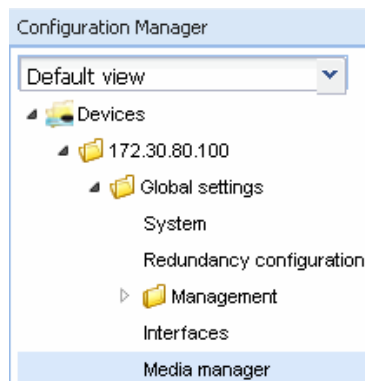
Wait time for invalid register

7. Click **Apply**.

Media manager

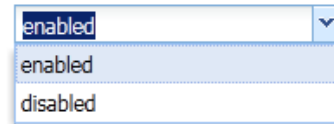
To configure protection against DDoS attacks from behind a NAT in Media manager:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. **Send syslog on demote To deny**—Select enabled from the drop-down list. The default value is disabled.

Send syslog on demote To deny



5. Click **Apply**.

CSV Filename Sequence Numbering

The basic local CDR file naming convention renders it impossible to determine if one or more files are missing from a list of file names, which is even more apparent when one views the list of files on a FTP/SFTP server.

Even when the file-rotation-time is enabled, viewing timestamps does not provide a definitive means for determining missing files, as an administrator could have changed the rotation time or maximum file size, causing fewer or more frequent rotations.

As a solution, the Net-Net SBC can append a sequence number to the CSV file name by enabling the **file-seq-number** parameter in the account config configuration element.

The sequence number is a 9 digit number padded with zeros where applicable. For example, sequence number 123 appears as 000000123 in a filename. While the sequence number will be consistent across failovers, it will be reset whenever the Net-Net SBC is power cycled.

With sequence numbers enabled, the format of a CSV file name will be:

`cdr<timestamp>-<9 digit sequence number>`

Using the format shown above, a directory listing will continue to show files in time-ascending order, even in the presence of a sequence number. For example:

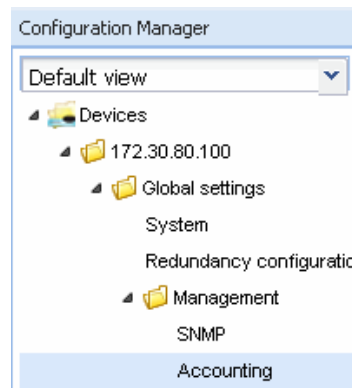
```
-rw-r--r-- 1 acme src 0 Nov 23 16:02 cdr200911212346-000000040
-rw-r--r-- 1 acme src 0 Nov 23 16:02 cdr200911212347-000000041
-rw-r--r-- 1 acme src 0 Nov 23 16:02 cdr200911212348-000000042
-rw-r--r-- 1 acme src 0 Nov 23 16:02 cdr200911212349-000000043
-rw-r--r-- 1 acme src 0 Nov 23 16:06 cdr200911212354-000000001
-rw-r--r-- 1 acme src 0 Nov 23 16:07 cdr200911212355-000000002
-rw-r--r-- 1 acme src 0 Nov 23 16:08 cdr200911212356-000000003
-rw-r--r-- 1 acme src 0 Nov 23 16:09 cdr200911212357-000000004
```

When this feature is enabled, the [a-j] suffix described in the Local CDR File Naming Convention section is disabled.

Adding Filename Sequence Numbers

To enable the Net-Net SBC to add filename sequence numbers:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Click the Push and file based tab.
5. **File sequence number**—Set this parameter to enabled for the Net-Net SBC to add sequence numbers to the end of the CDR CSV files.
6. Click **Apply**.

FTP/SFTP Server Upload Verification

When sending a CDR CSV file to a remote FTP or SFTP server, the file might not have been transferred completely. A failed file push will likely result in a partial file appearing on the server. By inspecting a partial file and its contents, it is not always possible to determine if it in fact is a partial file.

To ensure that CDR CSV files pushed to FTP or SFTP servers are saved in their entirety on the remote server, the Net-Net SBC uses an initial temporary file name for upload. Once the transfer completes, the Net-Net SBC renames the file to the permanent name with the appropriate FTP or SFTP command.

This behavior is configured in the the push-receiver portion of the accounting configuration by enabling the **temp remote file** parameter. When enabled, the Net-Net SBC transmits the file with a "temp-" prefix. Once the file transfer completes, the remote file is renamed to the final name. With this mechanism in place, the system operator can assume that all "tmp-" files are incomplete.

For example, a CDR CSV file is uploaded to an FTP or SFTP server as:

```
tmp-<filename-prefix>-cdr<timestamp>-<9 digit sequence number>
```

When the upload completes, it is renamed to:

```
<filename-prefix>-cdr<timestamp>-<9 digit sequence number>
```

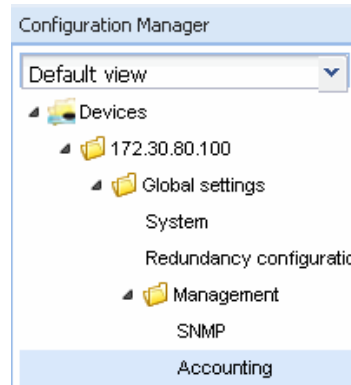
Server Requirements

- When connecting to a push receiver via FTP, ensure that the server supports the RNFR and RNT0 commands
- When connecting to a push receiver via SFTP, ensure that the server supports the rename command (packet type 18).

Configure Upload Verification

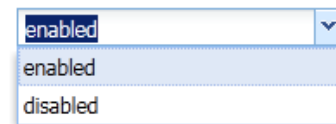
To configure the an account server for upload verification:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Click the Push and file based tab.
5. Scroll to CDR push servers and click the Edit button.
6. **temp-remote-file**—Select enabled from the combination box if you want the Net-Net SBC to upload CDR CSV files initially with a tmp-prefix and then rename the file after successful upload.

Temporary remote file



7. Click **Apply**.

Downgrade Warning

When downgrading system software from a version that supports CDR CSV file Sequence Numbers to a version that does not, there are a few guidelines:

- Disable file sequence numbers (file-seq-number = disabled) before you downgrade.
- Wait for all files with sequence numbers to be pushed off box, or retrieve them manually from `/randrv` on the Net-Net SBC.

Throttled Call Teardown to/from Session Agent

When a session agent goes OOS, the calls associated with that session agent are not immediately torn down. The calls are ultimately torn down either by the remote endpoint explicitly ending the call or by flowguard timers expiring.

You can configure the Net-Net SBC to explicitly tear down all calls associated with a session agent when it goes OOS. To mitigate any exceptionally high CPU load placed on the Net-Net SBC when it tears down all calls associated with the downed SA, internal processes will throttle the tear-down rate commensurate with the CPU load.

Calls that are already in progress before this option has been configured and activated are not subject to throttled teardown. For example, if this option is activated after 10 calls are in progress, then another 15 calls are added, all going through a single session agent. If the session agent goes OOS, only the second 15 calls are actively torn down.

Accounting Considerations

The Net-Net SBC writes the stop time that the calls are actually torn down and not the time when the SA went OOS for the disconnect time in the CDR. Media is stopped at the same time too.

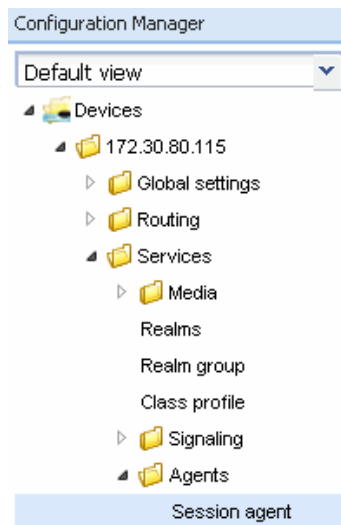
To indicate cases like this, where the call was terminated because the associated session agent went OOS, the Net-Net SBC includes the following combination of release/termination causes in the CDR:

```
Acct-Terminate-Cause = NAS-ERROR
Acme-Disconnect-Cause = 8
```

Configuring Throttled Tear-Down of Calls To/From an OOS SA

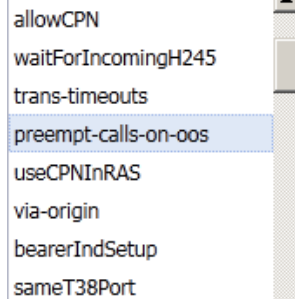
To configure throttled tear down of calls to/from an OOS SA:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the session agent you would like to use and click **Edit**.
4. Click Options. The Options window appears.

5. Select **preempt-calls-on-oos** in the Available Options list and click  to move the option to the Selected Options list.



6. Click OK.
7. Click Apply.

Support for Temporary Public User Identities Including Multi-SIM Scenarios

There is no configuration required for this feature. It is enabled automatically.

NN3800 Licensing Changes

There is no configuration required for this feature. It is enabled automatically.

NN3820 Licensing Changes

There is no configuration required for this feature. It is enabled automatically.

Network Management Control State

1963

You can show the state of a particular network management control by polling the apNetMgmtCtrlStatsTable. A new variable, apNetMgmtCtrlStatsState, has been added to this table to determine if a particular network management control is enabled or disabled.

SNMP Support

The ap-smgmt.mib MIB is expanded to include apNetMgmtCtrlStatsState.

MIB Table

The MIB table for network management control statistics is listed below:

```
apNetMgmtCtrlStatsTableOBJECT-TYPE
SYNTAXSEQUENCE OF ApNetMgmtCtrlStatsEntry
MAX-ACCESSnot-accessible
STATUScurrent
```


DESCRIPTION

"A table to hold statistics information on network management controls arranged into rows, indexed by unique names of the control rules. These are all read only."
 ::= { apSysMgmtMIBNetMgmtCtrlObjects 1 }

```

ApNetMgmtCtrlStatsEntry ::= SEQUENCE {
  apNetMgmtCtrlStatsNameDisplayString,
  apNetMgmtCtrlStatsTypeApNetMgmtCtrlType,
  apNetMgmtCtrlStatsIncomingTotalUnsigned32,
  apNetMgmtCtrlStatsRejectedTotalUnsigned32,
  apNetMgmtCtrlStatsDivertedTotalUnsigned32,
  apNetMgmtCtrlStatsIncomingCurrentUnsigned32,
  apNetMgmtCtrlStatsRejectedCurrentUnsigned32,
  apNetMgmtCtrlStatsDivertedCurrentUnsigned32,
  apNetMgmtCtrlStatsIncomingPeriodMaxUnsigned32,
  apNetMgmtCtrlStatsRejectedPeriodMaxUnsigned32,
  apNetMgmtCtrlStatsDivertedPeriodMaxUnsigned32,
  apNetMgmtCtrlStatsState                ApNetMgmtCtrlState
}

```

MIB Objects

The MIB objects for individual network management control rule statistics look like this:

```

apNetMgmtCtrlStatsNameOBJECT-TYPE
SYNTAX DisplayString (SIZE (1..255))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The name of the network management control the
following statistics are being calculated for"
::= {apNetMgmtCtrlStatsEntry 1}

```

```

apNetMgmtCtrlStatsTypeOBJECT-TYPE
SYNTAX ApNetMgmtCtrlType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The type of the specified network management
control, which can be gap-rate,gap-percent or
priority"
::= {apNetMgmtCtrlStatsEntry 2}

```

```

apNetMgmtCtrlStatsIncomingTotalOBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Total number of incoming calls that have
matched a destination identifier of the
specified network management control"
::= {apNetMgmtCtrlStatsEntry 3}

```

```

apNetMgmtCtrlStatsRejectedTotalOBJECT-TYPE

```

```

SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION
"Total number of incoming calls that have
been rejected by the specified network
management control"
::= {apNetMgmtCtrlStatsEntry 4}

apNetMgmtCtrlStatsDivertedTotalOBJECT-TYPE
SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION
"Total number of incoming calls that have
been diverted by the specified network
management control"
::= {apNetMgmtCtrlStatsEntry 5}

apNetMgmtCtrlStatsIncomingCurrentOBJECT-TYPE
SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION
"Number of incoming calls during the current
period that have matched a destination identifier
of the specified network management control"
::= {apNetMgmtCtrlStatsEntry 6}

apNetMgmtCtrlStatsRejectedCurrentOBJECT-TYPE
SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION
"Number of incoming calls during the current
period that have been rejected by the specified
network management control"
::= {apNetMgmtCtrlStatsEntry 7}

apNetMgmtCtrlStatsDivertedCurrentOBJECT-TYPE
SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION
"Number of incoming calls during the current
period that have been diverted by the specified
network management control"
::= {apNetMgmtCtrlStatsEntry 8}

apNetMgmtCtrlStatsIncomingPeriodMaxOBJECT-TYPE
SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION

```

```
"Maximum number of incoming calls during a period
that have matched a destinationidentifier of the
specified network management control"
::= {apNetMgmtCtrlStatsEntry 9}
```

```
apNetMgmtCtrlStatsRejectedPeriodMaxOBJECT-TYPE
SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION
"Maximum number of incoming calls during a period
that have been rejected by the specified network
management control"
::= {apNetMgmtCtrlStatsEntry 10}
```

```
apNetMgmtCtrlStatsDivertedPeriodMaxOBJECT-TYPE
SYNTAXUnsigned32
MAX-ACCESS read-only
STATUScurrent
DESCRIPTION
"Maximum number of incoming calls during a period
that have been diverted by the specified network
management control"
::= {apNetMgmtCtrlStatsEntry 11}
```

```
apNetMgmtCtrlStatsState OBJECT-TYPE
    SYNTAX      ApNetMgmtCtrlState
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The state of the specified network management
        control, which can be disabled or enabled"
    ::= {apNetMgmtCtrlStatsEntry 12}
```

Increase Session Limit and Remove 10X Reg Limit

There is no configuration required for this feature. It is enabled automatically.

Surrogate Agent Refresh on Invalidate

There is no configuration required for this feature. It is enabled automatically.

IPv6 SRTP MIKEY Key Exchange

There is no configuration required for this feature. It is enabled automatically.

Suppress DUP 200

There is no configuration required for this feature. It is enabled automatically.

Fragmented IP Packet Policing

You can set the maximum amount of bandwidth (in the max-untrusted-signaling parameter) you want to use for untrusted packets. However, because untrusted and fragmented packets share the same amount of bandwidth for policing, any flood of untrusted packets can cause the Net-Net SBC to drop fragment packets.

To prevent fragment packet loss, you can set the fragment-msg-bandwidth. When it is set to any value other than 0 (which disables it), the Net-Net SBC:

- Provides for a separate policing queue for fragment packets (separate from that used for untrusted packets)
- Uses this new queue to prevent fragment packet loss when there is a flood from untrusted endpoints.

When you set up a queue for fragment packets, untrusted packets likewise have their own queue—meaning also that the max-untrusted-signaling and min-untrusted-signaling values are applied to the untrusted queue.

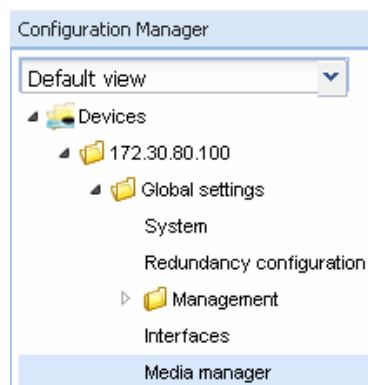
Enabling the fragmented message queue can also be used to shield untrusted traffic from a fragmentation attack.

Note: Enabling the fragmented message queue, i.e. changing the fragment msg bandwidth parameter from 0 to a non zero value requires a reboot to take effect. Changing the size of the queue once enabled does not require a reboot.

Enabling Fragmented IP Packet Policing

To enable fragmented IP packet policing:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.

3. Click **Add** to add the Media manager configuration in the content area.
4. Click the CPU policing and DoS tab at the top of the content area.
5. **Fragment rate limit (bytes per second)**—Enter the amount of bandwidth to use for the fragment packet queue. If you leave this parameter set to 0, then the Net-Net SBC will use the same queue for and share bandwidth between untrusted packets and fragment packets. The default value is zero (0). Valid policed values range between 32,000-10,000,000 Bytes/sec.

Fragment rate limit (bytes per second)

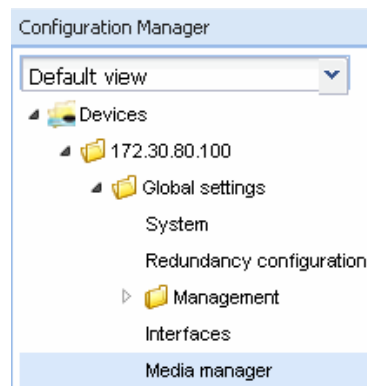
6. Click **Apply**.

Option Hairpin Released Flows

Hairpin Released Flows Option

To configure the hairpin released flows option:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to Options and click **Options**.
5. Select **hairpin-released-flows** in the Available options list and click  to move it to the Selected options list.
6. Click **OK**.
7. Click **Apply**.

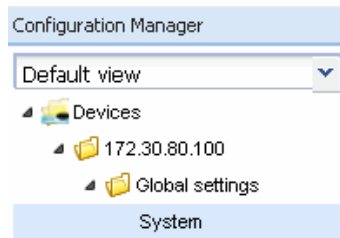
Configurable MTU Size for IPv4 and IPv6

Configuring MTU Size for IPv4 and IPv6

To configure `ipv4-mtu` and `ipv6-mtu`:

Note: IPv6 signaling MTU does not exist on the Net-Net 4200 series.

1. Expand the Configuration Manager slider > Global settings.



2. Click System. The System window appears in the content area.
3. **ipv4-mtu**—Enter a numerical value between 576 - 4096. The default value is 1500.
4. **ipv6-mtu**—Enter a numerical value between 576 - 4096. The default value is 1500.

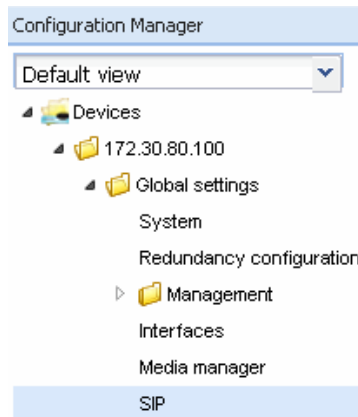
Click **Apply**.

P-CSCF PANI Enhancements

Configuring PANI Header for Trusted Domains

To configure PANI header only for trusted domains:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
 3. Click **Add** to add the SIP configuration in the content area.
 4. Scroll to Resource priority headers.
 5. **Allow PANI header only for trusted domains**—Click enabled in the drop down list to allow PANI headers for trusted domains. The default value is disabled.
- Allow PANI header only for trusted domains
6. Click **Apply**.

SNMP Extras

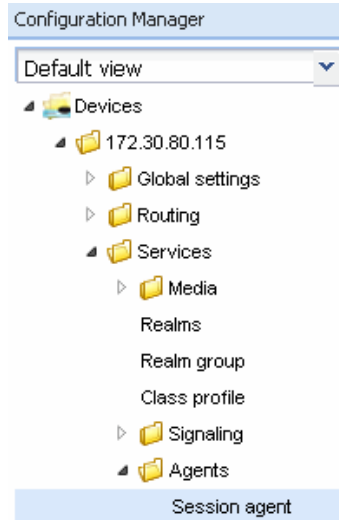
There is no configuration required for this feature. It is enabled automatically.

Asymmetric Payload Type

Configuring Asymmetric Payload Types

To configure asymmetric payload types:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the session agent you want to configure and click **Edit**.
4. Scroll to SIP options for this SA and click **Options**.
5. Select **rfc2833-allow-asymmetric-pt** in the Available options list and click  to move it to the Selected options list.
6. Click **OK**.
7. Click **Apply**.

Dual Stack Support

Previous Net-Net SBC releases supported both IPv4 and IPv6. These releases, however, required that IPv4 and IPv6 be segregated on separate VLANs when supported by the same physical interface. With the increasing number of IPv6 deployments, there are more Layer 2 Ethernet networks supporting both IPv4 and IPv6 deployments. Consequently, to adapt to changing topologies, the Net-Net SBC provides support for the definition of multiple network interfaces (IPv4 and IPv6) on the same physical interface and VLAN.

There is no configuration required for this feature. This feature is enabled automatically.

Media Stop Time VSA in CDRs

An accurate portrayal of a call's media stop time is important for billing accuracy. Calls are often terminated well after the media has stopped flowing for such reasons as network or equipment peculiarities.

Media Stop Time VSAs

To record the actual media stop time, the Net-Net Session Director writes the following four VSAs in CDR Stop Records:

```
Acme-Flow-Calling-Media-Stop-Time_FS1
Acme-Flow-Called-Media-Stop-Time_FS1
Acme-Flow-Calling-Media-Stop-Time_FS2
Acme-Flow-Called-Media-Stop-Time_FS2
```

These VSAs correspond to:

- calling side's media stop time - stream 1
- called side's media stop time - stream 1
- calling side's media stop time - stream 2
- called side's media stop time - stream 2

Media Stop Time Calculation

The granularity of the time at which the Net-Net Session Director's checks for media stream idleness, the actual media stop time, as inserted into a CDR is accurate to between 0 and +10 seconds.

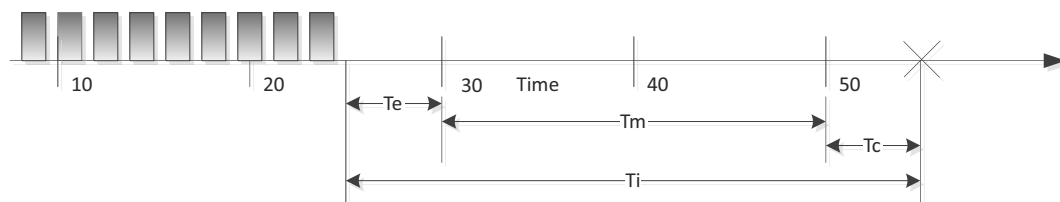
In the following diagram, media idleness monitoring is checked in 10 second time frames. Labeled time measurements are as follows:

Tc—Time between most recent idleness sample end and end-of-call time

Tm—2 complete idleness windows

Te—Time into the idleness window in which the call's media stopped. This is also the error amount of the recorded media stop time.

Ti—The actual time between the end of media and the end of call.



Tm and Tc are known. The Net-Net Session Director also knows that the media ended between 20 and 30 seconds, but the actual time, 10 - Te into the frame is unknown. Thus, the time recorded in the CDR is quantized up to the end of the media stop frame at 30 seconds. This time, as written to the CDR, must be interpreted with possible error of $0 \leq Te < 10$ seconds.

HA Caveat

When a switchover occurs between media stop time and end of call, the media stop time written to the CDR is the failover time.

Diameter Rf Charging Buffering and Storage

About Buffering

Diameter Rf Charging, Buffering, and Storage enables the Net-Net SBC to buffer all accounting requests (ACR) in memory for a configurable number of ACRs. The buffer holds a minimum of 15 minutes of ACRs under busy-hour load conditions. For example, based on intended traffic, the buffer would hold 54 calls-per-second which equals approximately 150,000 records. The Net-Net SBC sends an SNMP trap when accounting records begin to drop from the buffer due to an overflow condition. Subsequently, a clearing SNMP trap is sent once the fault condition is removed.

About Storage

The Net-Net SBC maintains storage to temporarily store accounting records to a charging collection function (CCF) link, in case of a failure or congestion exists on the Net-Net SBC. In this scenario, the Net-Net SBC can store a minimum of 3 days-worth of accounting records. All ACRs can remain in storage for a configurable amount of time, and for a minimum of 3 days under normal traffic-load conditions.

There are two configurable options for storing ACRs:

- Store all ACRs generated by the Net-Net SBC
- No ACRs in storage

Monitoring Storage Space

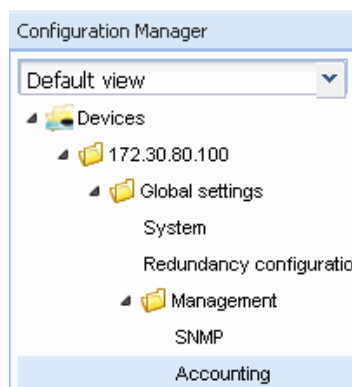
Disk storage space monitoring can be done on the total drive, or by disk partition. You can monitor storage space using the one of the following methods:

- Command line interface (CLI)
- SNMP management information base (MIB)
- Historical data records (HDR)

Configuring Diameter Rf Buffering and Storage

To configure Diameter Rf buffering and storage size:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.

4. Click the Accounting tab.
5. **Diameter send throttle**—Enter the value for the diameter send throttle. This parameter applies to the Diameter accounting interface only. The valid range is 2 - 20. The default value is 20.

Diameter send throttle

6. **Message queue size**—Enter the message queue size. This parameter applies to both RADIUS and Diameter accounting interfaces. The valid range is 5000 - 150000. The default value is 5000.

Message queue size

7. Click **Apply**.

Diameter Attribute ID Range Selection

With this feature, you can customize the Attribute Value Pairs (AVPs) sent to the charging collection function (CCF) by specifying the AVPs to include in the Diameter Rf accounting records. All AVPs are included, if not specified. The default setting (blank field) is to send all AVPs.

The parameter, **Diameter attribute ID range**, controls which AVPs to include in the Diameter Rf accounting records, and applies to the Diameter Rf interface only. The value range for this parameter is 1 through 876.

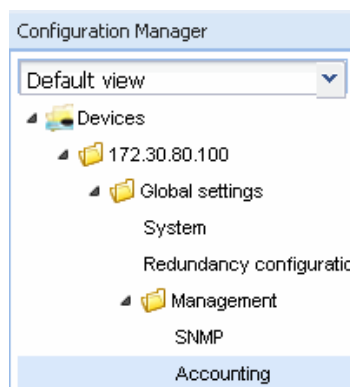
There are two conditions for grouped AVPs:

- If a group AVP attribute ID is not listed in the range, the AVPs that belong to this group are not present in the Diameter Rf accounting records, regardless of whether or not they are listed in the range.
- If a group AVP attribute ID is listed in the range, the AVPs that belong to this group are present in the Diameter Rf accounting records, regardless of whether or not they are listed in the range.

Configuring Diameter Rf Attribute ID Range

To configure the Diameter attribute ID range:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.

3. Press **Add** to add the accounting configuration in the content area.
4. Click the Accounting tab.
5. Scroll to Diameter attribute ID range.
6. Click **Add**. The Add Diameter attribute ID range dialog box appears.
7. **Diameter attribute ID range**—Enter Diameter Rf accounting records ranging from 1-876. This parameter controls which AVPs are included in the Diameter Rf accounting records. The default value is blank, or **disabled**, which includes all AVPs.

Diameter attribute ID range

8. Click **OK**.
9. To add additional Diameter attribute ID ranges, repeat steps 6 through 8.
10. Click **Apply**.

Session Plug-In Language (SPL)

Session Plug-In Language (SPL) provides a means for Acme Packet to craft solutions and features to unique problems and deploy them in a portable plugin-type software package. SPL plugins are uploaded to the Net-Net SBC, marked to be executed, and then perform a feature-like function as expected. SPL only works for SIP messaging. You may only run signed SPL files on your Net-Net SBC available directly from Acme Packet, Inc.

Upon boot, the Net-Net SBC compiles all scripts in the `/code/spl` directory that are configured in the `spl config` configuration element. If there is an error during parsing tile SPL files, it is written to the `log.sipd` and the script is not loaded. Scripts are loaded in the order in which they are configured in the `spl plugins` configuration element.

SPL Packages act identically to SPL plugins but contain multiple plugins in one file. When a package file is configured in the **name** parameter of the `spl plugins` configuration element, the Net-Net SBC will load all SPL plugins contained in that package. You may also configure the Net-Net SBC to execute a single plugin contained within the package with the syntax `package-name:plugin-name`. You can omit the `.pkg` extension when configuring the Net-Net SBC to load one plugin from a package.

Enabling SPL Plugins

Enabling SPL plugins is a three step process.

1. Copy the SPL plugins to a Net-Net SBC.
2. Configure the Net-Net SBC to recognize and run the plugins.
3. Command the Net-Net SBC to execute the plugins.

Uploading SPL Plugins

SPL plugins must be manually FTPed to the Net-Net SBC's `/code/spl` directory with use any CLI or GUI-based FTP or SFTP application. The Net-Net SBC's FTP server, if enabled may be reached from the system's `wancom` or `eth0` management physical interface. For example:

```

C:\Desktop>ftp 172.30.46.10
Connected to 172.30.46.10.
220 ACME SYSTEM FTP server (VxWorks 6.4) ready.
User (172.30.46.10:(none)): user
331 Password required for user.
Password:
230 User user logged in.
ftp> cd /code/spl
250 CWD command successful.
ftp> put HelloWorld.spl
200 PORT command successful.
150 Opening ASCII mode data connection for '/code/spl/HelloWorld.spl'.
226 Transfer complete.
ftp: 317 bytes sent in 0.04Seconds 8.13Kbytes/sec.
ftp> bye
221 Goodbye.

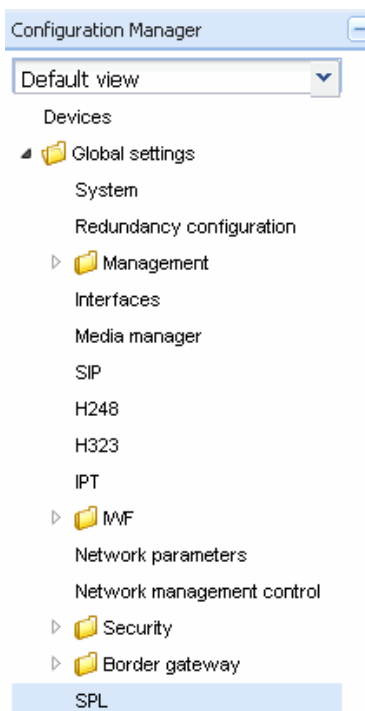
```

Configuring SPL Plugins

All SPL plugin files that you intend to run must be configured in the *spl plugins* configuration element. The Net-Net SBC executes the plugin files in the order in which they were configured.

To add an SPL Plugin or SPL Package to the configuration:

1. Expand the Configuration Manager slider > Global settings.
2. Click SPL.



3. Plugins—Click **Add**. The Add Plugins dialog box appears.
4. **Plugin**—Enter the name of a plugin file in the `/code/spl` directory that you wish the Net-Net SBC to execute. You must enter a unique name, which consists of alphanumeric characters. You can enter:

- The name of the SPL Package file in the name parameter, for example: **SPL_PACKAGE.PKG**
- A single SPL Plugin that exists in a package file, for example: **SPL_PACKAGE:MODIFY_HEADER**

*Plugin

SPL_package.pkg

5. Click **Apply**. A Success message appears.
6. Click **OK**. The plugin is added to the Plugins table.

SPL Parameter Configuration

SPL Plugins may create the **spl-options** parameter in either the Session agent, SIP interface, Realm, or SPL configurations. The **spl-options** parameter appears in Net-Net Central after an SPL plugin that creates the parameter is executed. The **spl-options** parameter will not necessarily appear in all four (or any) configuration elements. Where and when to configure the **spl-options** parameter is discussed in each plugin's specific documentation.

Executing SPL Files

There are three ways to execute SPL files:

1. Perform a **Save** and **activate configuration**.
2. Reboot the system (after a **Save configuration**).
3. Execute the **reset spl** command—all configured SPL files are refreshed by with the **reset spl** command. You can also refresh a specific file by typing **reset spl <spl-file>**.

Note: Acme Packet suggests that scripts are only refreshed during system downtime.

If an SPL file exists in the `/code/spl` directory, but is not configured in the **spl-files** parameter, it will be ignored when the Net-Net SBC loads all SPL plugins. You may still manually load an SPL file directly with the **reset** command.

In this case, the operator must remember that a manually-loaded SPL file will not be automatically loaded on the next reboot.

Synchronizing SPL Files Using the ACLI

Note: Synchronizing SPL files cannot be performed using Net-Net Central. This procedure must be done using the ACLI.

When running in an HA configuration, both the active and the standby system must have the same version of the running SPL plugins installed. To facilitate configuring the standby system, the **synchronize spl** ACLI command has been developed. Executing this command without any arguments copies all files in the `/code/spl` directory from the active system to the to the standby Net-Net SBC overwriting any existing files with the same name.

By adding the specific filename as an argument to the **synchronize spl** command, the individual, specified scripts are copied between systems. For example:

```
ACMEPACKET#synchronize spl HelloWorld.spl
```

The **synchronize spl** command can only be executed from the active system in a HA pair. There is no means to automatically synchronization SPL files during a save and activate of the SBC.

SPL Signature State

Upon executing **show spl <task>**, the ACLI displays SPL file information including the signature which will be in one of three states:

1. not signed
2. signed and valid
3. signed but invalid

Deleting SPL Plugin Files

Deleting files from `/code/spl` must be performed via FTP; there is no means to delete files from Net-Net Central or from the ACLI.

SPL Log Types

The SPL log messages can often be found in the respective task's log file when that task is set to DEBUG level.

Old File Remover

You can enable the Net-Net SBC to clean up old files automatically by configuring the exact directories in which those files reside and the time at which they should be purged. To enable this capability, you use the **path** and **cleanup time of day** settings in the system configuration.

For each directory you want cleaned, you identify the directory path as explicitly as possible (because subdirectories are not cleaned) and the age of the files to remove from the directory. The configuration allows you to turn each file remover on and off, setting the **state** parameter to **enabled** for on and **disabled** for off. You also set the time of day, in hours and minutes in local time, to identify when the system performs the clean-up.

Specifics and Caveats

This section calls out special safeguards, limitations, and cautions to note before you enable file removal.

Protected Directories

As a preventive measure, the Net-Net SBC does not permit you to remove files from the following directories (and their subdirectories) that are vital to correct system functioning:

- /code—Contains essential system files.
- /boot—Contains essential system files.
- /ramdrv/logs—Contains all logs, including: CLI audit logs, the **support info** to-file. However, you can still use the log cleaner function configurable under the system configuration's **options** parameter.
- /ramdrv/collect—Contains all HDR files.

If you try to configure any of these directories, the system will deem your entry invalid.

Partially Protected Directories

The directories listed below cannot be cleaned themselves, though you can clean their subdirectories:

- The CDR file path—Configured in the accounting configuration's **file-path** parameter. If configured, cleaning this directory would interfere with proper operation of the FTP push capability you can establish as part of your accounting configuration; it would interfere with file rotation and deletion. The Net-Net SBC will return an warning when you verify your configuration if you mark this file path as one to clean. This warning informs you that the Net-Net SBC will not be cleaned.
- /ramdrv—Apart from its subdirectories, this directory contains files that could be vital to proper system functioning. If you try to enter the main /ramdrv directory, the system will deem your entry invalid.

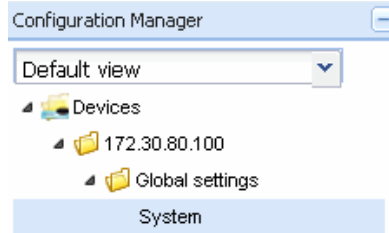
Configuring Old File Purging

The following sections show you how to configure the Net-Net SBC to clean up old files daily at a specified time.

Setting Daily Local Time

To set the daily local time at which the Net-Net SBC will clean up files:

1. Expand the Configuration Manager slider > Global settings.
2. Click System.



3. Scroll to **Cleanup-time-of-day**.
4. **Cleanup-time-of-day**—Change this parameter from its default (00:00) to the local time when you want the Net-Net SBC to clean up files every day. You enter the time as HH:MM, where HH represents hours and MM represents minutes.

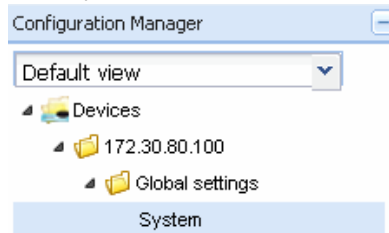
Cleanup time of the day

5. Click **Apply**.

Identifying Directories for Cleanup

To identify the directories the Net-Net SBC will clean up:

1. Expand the Configuration Manager slider > Global settings.
2. Click System.



3. Scroll to Directory cleanup.
4. Click **Add**. The Add Directory cleanup dialog box appears.
5. **Path**—Enter the name of the directory path where you want the Net-Net SBC to perform file clean-up. Remember that subdirectories are not examined or cleaned. This parameter is required and is empty by default.

Add Directory cleanup
✕

*Path

pathA

(Unique required)

Apply

Cancel

6. Click **Apply**. A Success dialog box appears.
7. Click **OK**.
8. The path is added to the Directory cleanup table.

9. Click the directory cleanup path you want to configure and click **Edit**.
10. **State**—To turn on daily clean-up for the directory path you specified in the **path** parameter, leave this set to its default: **enabled**. If you want to turn off clean-up for the specified directory, click **disabled** in the drop-down list.
11. **Age**—Enter the age of the files to be deleted in number of days. For example, if you want the Net-Net SBC to clean all files that are three days old, you set this parameter to **3**. Files older than this number of days will be purged. The default and maximum value is **30** (days), and the minimum value is **1** (day).

State	enabled ▼
Age	30

12. Click **Apply**.

Temporary File Naming for an Open CDR File

As of Net-Net OS Release S-C(X)6.0.0M7, the Net-Net SBC uses a temporary naming convention that makes it easier for you to retrieve CDR files you want.

Before this release was introduced, the Net-Net SBC used the same naming format for all CDR files: **cdrYYYYMMDDHHMM[a-j]**. If this is the naming convention you still want to use, you can do so simply by adding the **disable-temp-file** option to your accounting configuration. This mode offers no means of differentiating a file to which the Net-Net SBC is writing information from any other closed file(s).

If you decide to use the new default behavior, then you will now see a the **temp-** prefix added to the file format. So the file format for the temp file is: **temp-cdrYYYYMMDDHHMM[a-j]**. The prefix helps you differentiate the file that is currently open from the other CDR files you encounter; this is the file to which the Net-Net SBC is currently writing information and is open. As soon as the file is closed during rotation, the **temp-** disappears and the file bears only name in the **cdrYYYYMMDDHHMM[a-j]**. In other words, files in the **cdrYYYYMMDDHHMM[a-j]** are closed files.

Without this differentiation, it is possible for you to retrieve different versions of the same file and to even do so more than once. In addition, without the **temp-** differentiation, the Net-Net SBC FTP server is liable to return error messages when move or delete operations occur. These occurrences can trigger false alarms and are not consistent with other vendors' products.

Operational Details

This section offers details of Net-Net SBC operations that effect temporary CDR file naming.

- **Reboot**—A system reboot can happen unexpectedly, or might be caused by your intentionally rebooting the system using the ACLI **reboot** command. When a reboot occurs, Net-Net SBC closes the CDR file that was most recently opened (before the reboot) and names it according to the **cdrYYYYMMDDHHMM[a-j]** convention. It also opens a new file, which bears the **temp-** differentiation.
- **Activating a configuration**—If temporary CDR naming is enabled before and after you use the **activate-config** command, then the last opened file will be closed and have the **cdrYYYYMMDDHHMM[a-j]** name format. The Net-Net SBC also opens a new file with the **temp-** prefix to which it will write data.

In the case where temporary CDR naming is enabled before you activate a configuration and disabled after it, the last open file is named according to the **cdrYYYYMMDDHHMM[a-j]** name format. The new file to which the Net-Net SBC will write data is also in the **cdrYYYYMMDDHHMM[a-j]** name format. In other words, the Net-Net SBC does not use the **temp-** prefix designation at all.

In the case where temporary CDR naming is disabled before you activate a configuration and enabled after it, the Net-Net SBC closes the most recently opened file, which must have been in the **cdrYYYYMMDDHHMM[a-j]** name format. The Net-Net SBC also opens a new file with the **temp-** prefix to which it will write data.

- Changing the accounting configuration's administrative state—When you disable the accounting configuration, the Net-Net SBC renames the most recently opened file with the **temp-** prefix to the **cdrYYYYMMDDHHMM[a-j]** name format.

HA Considerations

The considerations in this section describes the Net-Net SBC's behavior when CDR output redundancy is enabled or disabled. You set CDR output redundancy in the accounting configurations **CDR output redundancy** parameter.

- Enabled—When you enable CDR output redundancy, both the Active and Standby systems rotate files. During CDR file rotation, if either the Active or the Standby rotates a file with the **temp-** prefix, the prefix disappears and the file name appears in the **cdrYYYYMMDDHHMM[a-j]** name format.

The Active and the Standby systems always have the same files, including the CDR file with the **temp-** prefix. So the file exists on both systems.

- Disabled—When you have disables CDR output redundancy and switchover happens for any reason, it is key that there are no residual files with the **temp-** prefix. For this reason, the Net-Net SBC handles the situation as follows:

Becoming Active—When it transitions from Standby to Active, a Net-Net SBC checks for any files with the **temp-** prefix, closes the file if it is open, and renames it according to the **cdrYYYYMMDDHHMM[a-j]** name format. These actions means that the file is not only renamed, but that it is also rotated. Rotation triggers the creation of a new CDR file with the **temp-** prefix to use for new CDR data.

Becoming Standby—When it transitions from Active to Standby, a Net-Net SBC closes the open **temp-** prefix file and renames it according to the **cdrYYYYMMDDHHMM[a-j]** name format. Rotation creates a new **temp-** prefix file on the Standby, which remains empty until it transitions back to the Active state.

Caveats

As described above, when the system reboots for any reason or when you issue an **Save & Activate**, the Net-Net SBC checks for CDR files with the **temp-** prefix and renames to the usual **cdrYYYYMMDDHHMM[a-j]** format.

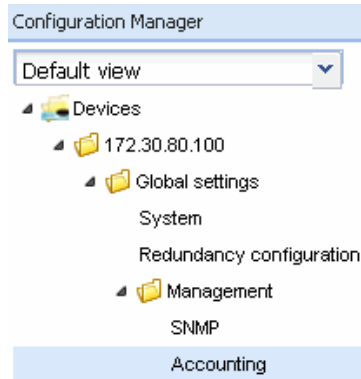
However, if you change the accounting configuration's file-path value and subsequently the system either reboots or you activate your configuration, the Net-Net SBC will be unable to check for files with the **temp-** prefix in the old file path. And so it will also be unable to rename them. The Net-Net SBC checks the new path only.

Disabling Temporary CDR File Naming

Since Release S-C(x)6.0.0M7 and later use this capability by default, this section offers instructions for how to turn off temporary CDR file naming if you do not want to use it.

To disable temporary CDR file naming:

1. Expand the Configuration Manager slider > Global settings > Management.
2. Click Accounting.



3. Click the Accounting tab.
4. Scroll to Options. The Options selector dialog box displays.
5. Click **Options**.
6. Click **disable-temp-file** in the Available options list to select it.
7. Click **>>>** to move the option to the Selected options list.
8. Click **OK**.

This value turns off the temporary CDR file naming the Net-Net SBC, so it does not use the **temp-** prefix for open file. Instead, all files follow the **cdrYYYYMMDDHHMM[a-j]** name format.

9. Click **Apply**.

Note: By pressing **Apply**, you are applying your changes to the Net-Net Central database. You must perform a **Save & Activate** to apply the changes to your device.

Introduction

This chapter explains how to configure realms and nested realms.

A realm is a logical definition of a network or groups of networks made up in part by devices that provide real-time communication sessions comprised of signaling messages and possibly media flows. These network devices might be call agents, softswitches, SIP proxies, H.323 gatekeepers, IP PBXs, etc., that are statically defined by IP addresses. These network devices might also be IP endpoints: SIP phones, IADs, MAs, media gateways, etc., that are defined by an IP address prefix.

Realms support bandwidth-based call admission control and QoS marking for media. They are the basis for defining egress and ingress traffic to the Net-Net SBC—which supports the Net-Net SBC's topology hiding capabilities.

This chapter also explains how to configure media ports (steering pools). A steering pool exists within a realm and contains a range of ports that have a common address (for example, a target IP address). The range of ports contained in the steering pool are used to steer media flows from one realm, through the Net-Net SBC, to another.

Finally, in this chapter you can learn about TOS/DiffServ functionality for realm-based packet marking by media type.

Overview

Realms are a logical distinction representing routes (or groups of routes) reachable by the Net-Net SBC and what kinds of resources and special functions apply to those routes. Realms are used as a basis for determining ingress and egress associations to network interfaces, which can reside in different VPNs. The ingress realm is determined by the signaling interface on which traffic arrives. The egress realm is determined by the following:

- Routing policy—Where the egress realm is determined in the session agent configuration or external address of a SIP-NAT
- Realm-bridging—As applied in the SIP-NAT configuration and H.323 stack configurations
- Third-party routing/redirect (i.e., SIP redirect or H.323 LCF)

Realms also provide configuration support for denial of service (DoS)/access control list (ACL) functionality. For more information about the Net-Net SBC's DoS/ACL capabilities and configuration, refer to this guide's *Security* chapter.

Realms can also be nested in order to form nested realm groups. Nested realms consist of separate realms that are arranged within a hierarchy to support network architectures that have separate backbone networks and VPNs for signaling and media. This chapter provides detailed information about nested realms after showing you how to configure realms on your Net-Net SBC.

About Realms and Network Interfaces

All realms reference network interfaces on the Net-Net SBC. This reference is made when you configure a list of network interfaces in the realm configuration.

You configure a network interface to specify logical network interfaces that correspond existing physical interfaces on the Net-Net SBC. Configuring multiple network interfaces on a single physical interface creates a channelized physical interface, a VLAN. VLANs, in turn, allow you to reuse address space, segment traffic, and maximize bandwidth.

In order to reach the realms you configure, you need to assign them network interfaces. The values you set for the name and port in the network interface you select then indicate where the realm can be reached.

About the SIP Home Realm

The realm configuration is also used to establish what is referred to as the SIP home realm. This is the realm where the Net-Net SBC's SIP proxy sits.

In peering configurations, the SIP home realm is the internal network of the SIP proxy. In backbone access configurations, the SIP home realm typically interfaces with the backbone connected network. In additions, the SIP home realm is usually exposed to the Internet in an HNT configuration.

Although you configure a SIP home realm in the realm configuration, it is specified as the home realm in the main SIP configuration by the home realm identifier parameter. Specifying the SIP home realm means that the Net-Net SBC's SIP proxy can be addressed directly by connected entities, but other connected network signaling receives layer 3 NAT treatment before reaching the internal SIP proxy.

For more information about SIP functionality and features, refer to this guide's *SIP Configuration* chapter.

About Realms and Other Net-Net SBC Functions

Realms are referenced by other configurations in order to support this functionality across the protocols the Net-Net SBC supports and to make routing decisions. Other configurations' parameters that point to realms are:

- SIP configuration: home realm identifier, egress realm identifier
- SIP-NAT configuration: realm identifier
- H.323 stack configuration: realm identifier
- MGCP configuration: private realm, public realm
- Session agent configuration: realm identifier
- Media manager: home realm identifier
- Steering ports: realm identifier
- Static flow: in realm identifier, out realm identifier

Configuring Realms

Realm configuration is divided into the following functional areas, and the steps for configuring each are set out in this chapter: identity and IP address prefix, realm interfaces, realm service profiles, QoS measurement, QoS marking, address translation profiles, and DNS server configuration.

Before You Configure

Before you configure realms, you want to establish the physical and network interfaces with which the realm will be associated.

- Configure a physical interface to define the physical characteristics of the signaling line.
- Configure a network interface to define the network in which this realm is participating and optionally to create VLANs.

If you want to use QoS, you should also determine if your Net-Net SBC is QoS enabled. Remember that you will also use this realm in other configurations to accomplish the following:

- Set a signaling port or ports at which the Net-Net SBC listens for signaling messages.
- Configure sessions agents to point to ingress and egress signaling devices located in this realm in order to apply constraint for admission control.
- Configure session agents for defining trusted sources for accepting signaling messages.

Configuration Overview

This section contains a configuration overview:

- Identity and IP address prefix— Configure a realm name (a unique identifier) and an IP address prefix and subnet mask.

The IP address and subnet mask establish a set of matching criteria for the realm, and distinguishes between realms that you assign to the same network interface.

- Realm interfaces—The realm points to one or more network interfaces on the Net-Net SBC. For more information, refer to this chapter's *About Realms and Network Interfaces* section. For information about configuring network interfaces and VLAN support, refer to this guide's *System Configuration* chapter.
- Realm service profile—The parameters you configure to establish the realm service profile determine how bandwidth resources are used and how media is treated in relation to the realm. Bandwidth constraints set for realm service profiles support the Net-Net SBC's admission control feature. For further information about this feature, refer to this guide's *Admission Control and QoS* chapter.

Peer-to-peer media between endpoints can be treated in one of three different ways:

- Media can be directed between sources and destinations within this realm on this specific Net-Net SBC. Media travels through the Net-Net SBC rather than straight between the endpoints.
- Media can be directed through the Net-Net SBC between endpoints that are in different realms, but share the same subnet.
- For SIP only, media can be released between multiple Net-Net SBCs.

To enable SIP distributed media release, you must set the appropriate parameter in the realm configuration. You must also set the SIP options parameter to media-release with the appropriate header name and header parameter information. This option defines how the Net-Net SBC encodes IP address and port information for media streams described by, for example, SDP.

- QoS measurement—Refer to this guide’s *Admission Control and QoS* chapter for more information about enabling QoS measurements on your Net-Net SBC. This chapter provides detailed information about when to configure the **qos-enable** parameter. If you are not using QoS or a QoS-capable Net-Net SBC, then you can leave this parameter set to **disabled** (default).
- QoS marking—QoS marking allows you to apply a set of TOS/DiffServ mechanisms that enable you to provide better service for selected networks.

You can configure a realm to perform realm-based packet marking by media type, either audio/voice or video.

The realm configuration references a set of media policies that you configure in the media policy configuration. Within these policies, you can establish TOS/DiffServ values that define an individual type (or class) of service, and then apply them on a per-realm basis. In the media profiles, you can also specify:

- One or more audio media types for SIP and/or H.323
- One or more video types for SIP and/or H.323
- Both audio and video media types for SIP and/or H.323
- Address translation profiles—Refer to this guide’s *Number Translations* chapter for realm-specific information about using address translations on your Net-Net SBC. If you are not using this feature, you can leave the **in-translationid** and **out-translationid** parameters blank.
- DNS servers—You can configure DNS functionality on a per-network-interface basis, or you can configure DNS servers to use per realm. Configuring DNS servers for your realms means that you can have multiple DNS servers in connected networks. In addition, this allows you to specify which DNS server to use for a given realm such that the DNS might actually be in a different realm with a different network interface.

This feature is available for SIP and MGCP only.

- DoS/ACL configuration—Refer to this guide’s *Security* chapter for realm-specific information about using DoS/ACL functionality on your Net-Net SBC. If you are not using this functionality, you can leave the parameters at their default values.
- Admission control configuration—You can set admission control based on bandwidth for each realm by setting the **max-bandwidth** parameter for the realm configuration. Details about admission control are covered in this guide’s *Admission Control and QoS* chapter.

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by	admin@console
Last modified date	2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2_UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

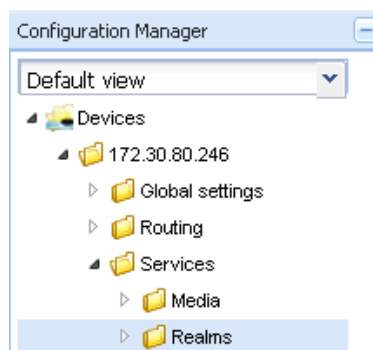
Configuring Realms

This section describes how to configure realms on your Net-Net SBC from the Net-Net Central.

Creating the Realm

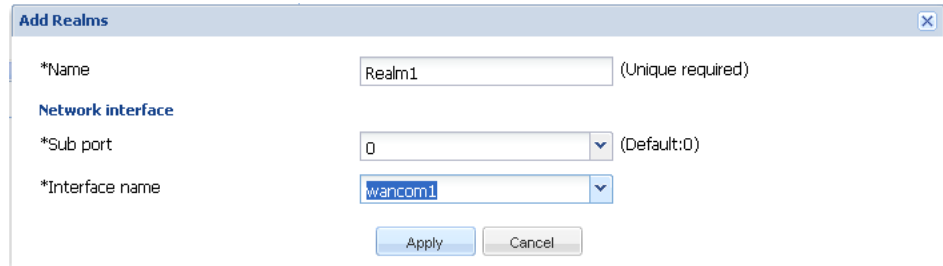
To create a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click **Add**. The Add instance dialog box appears.
4. **Name**—Enter the name of the new realm you are creating.
5. Click **Apply**.

This parameter uniquely identifies the realm. You will use this parameter in other configurations when asked for a realm identifier value.



The 'Add Realms' dialog box is shown. It has a title bar 'Add Realms' with a close button. Inside, there are three fields: '*Name' with the value 'Realm1' and a note '(Unique required)'; '*Sub port' with a dropdown menu showing '0' and a note '(Default:0)'; and '*Interface name' with a dropdown menu showing 'wancom1'. At the bottom are 'Apply' and 'Cancel' buttons.

The newly created realm appears in the Realm window.

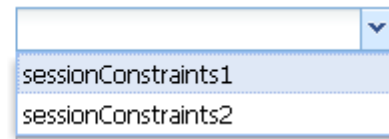
Configuring the Realm

1. Select the newly created realm and click **Edit**. The Edit Realm window appears in the Net-Net Central display pane.
2. **Session constraint name**—Select the name of a constraint you want to use for this realm from the combination box.
3. **IP address prefix**—Enter the IP address prefix and subnet mask combination to set the criteria the Net-Net SBC uses to match packets sent or received on the network interface associated with this realm. This matching determines the realm, and subsequently what resources are used for that traffic.

This parameter must be entered in the correct format where the IP address comes first and is separated by a slash (/) from the subnet mask value. For example, 172.16.0.0/24.

If you leave this parameter set to its default, 0.0.0.0/0, then all addresses will match.

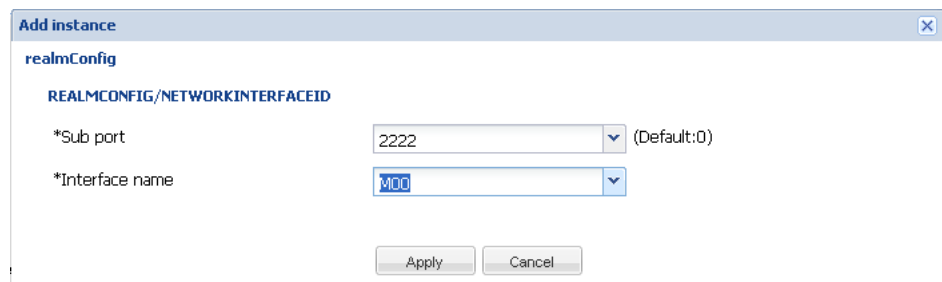
Session constraint name



A dropdown menu for 'Session constraint name' is shown. It has a downward arrow button. The menu is open, showing two options: 'sessionConstraints1' and 'sessionConstraints2'.

4. Click **Add** in the Network interfaces section of the Realm window. The Add instance dialog box appears.

This is the interface and subport through which this realm can be reached by ingress traffic, and through which this traffic exits the system as egress traffic.



The 'Add instance' dialog box is shown. It has a title bar 'Add instance' with a close button. Inside, there is a label 'realmConfig' and a sub-label 'REALMCONFIG/NETWORKINTERFACEID'. There are two fields: '*Sub port' with a dropdown menu showing '2222' and a note '(Default:0)'; and '*Interface name' with a dropdown menu showing 'VOC'. At the bottom are 'Apply' and 'Cancel' buttons.

5. **Sub port**—Select the VLAN identifier from the combination box for the network interface to which you are assigning this realm.

6. **Interface name**—Select a physical interface from the combination box.
7. Click **Apply** to continue.
8. **Bandwidth limit (kilobits per second)**—Set the maximum bandwidth resources available for this realm; the default is 0.

*Bandwidth limit(kilobits per second),Session admission control

9. **Within this realm serviced by this Net-Net system only**—Set this field to enabled to allow media steering when endpoints within the same realm are communicating.
10. **Within different realms in the same subnet serviced by this Net-Net system only**—Set this field to enabled to set the behavior of media steering when endpoints located in different realms, but that are connected to the same network interface, are communicating.
11. **Within this realm serviced by different Net-Net systems (SIP only)**—Set this field to enabled to include multi-system release information in SIP requests sent to this realm.

Manage media between endpoints

*Within this realm serviced by this Net-Net system only

*Within different realms in same subnet serviced by this Net-Net system only

*Within this realm serviced by different Net-Net systems (SIP only)

12. Set the number translations that you want to apply to this realm. Refer to this guide's *Number Translations* section for realm-specific information about using address translations on your Net-Net SBC. If you are not using this feature, you can leave the **in-translationid** and **out-translationid** parameters blank.
13. **Inbound**—Set an inbound number translation profile from the combination box.
14. **Outbound**—Set an outbound number translation profile from the combination box.

Address translation profiles

Inbound

Outbound

15. **Use DNS Server in this realm**—Select the realm whose network interface's DNS server should be used to resolve this realm's FQDN lookup.

If you do not configure this parameter, then the realm will use the DNS information configured in its associated network interface.

Use DNS Server in this realm

16. Click **Apply**.

RTP-RTCP UDP Checksum Generation

You can configure your Net-Net SBC to generate a UDP checksum for RTP/ RTCP packets on a per-realm basis. This feature is useful in cases where devices performing network address translation (NATs) do not pass through packets with a zero checksum from the public Internet. These packets do not make it through the NAT even if they have the correct to and from IP address and UDP port information. When you enable this feature, the Net-Net SBC calculates a checksum for these packets and thereby enables them to traverse a NAT successfully.

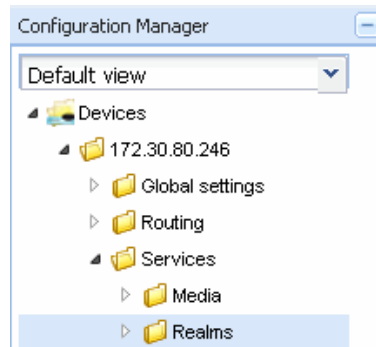
If you do not enable this feature, the Net-Net SBC will not generate a checksum for RTP or RTCP packets if their originator did not include one. If a checksum is already present when the traffic arrives at the Net-Net SBC, the system will relay it.

You enable this feature on the outbound realm.

Configuring UDP Checksum Generation

To enable UDP checksum generation for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a Realm and click **Edit**. The Edit Realm window appears.
4. **Generate UDP checksum on outbound media packets**—Select enabled from the combination box if you want the Net-Net SBC to generate a UDP checksum for this outbound realm.

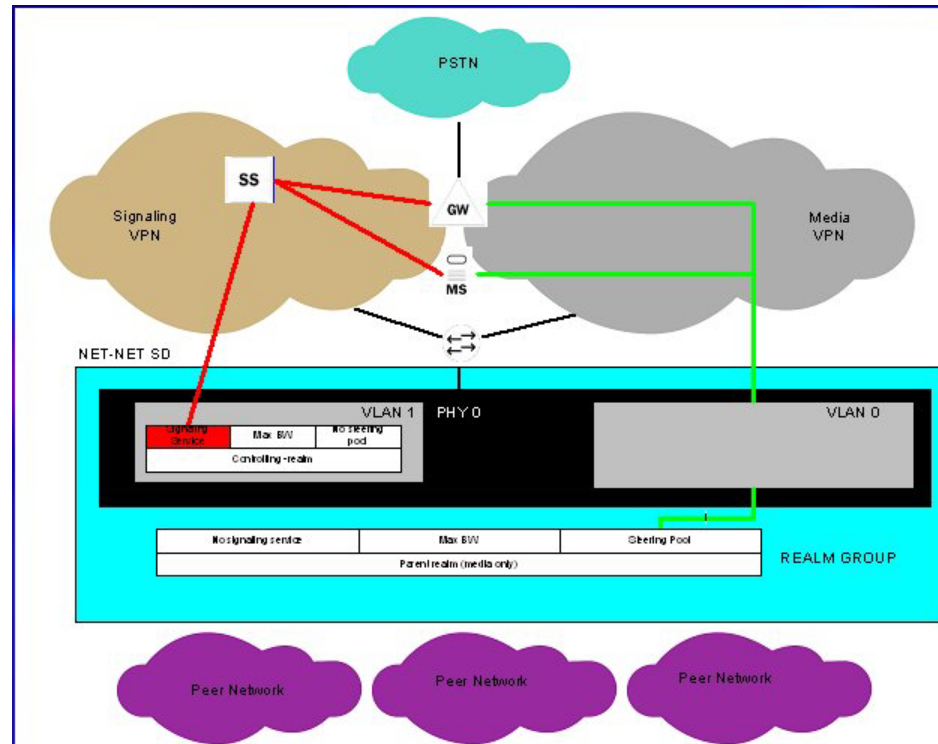
Generate UDP checksum on outbound media packets

5. Click **Apply**.

Nested Realms

Configuring nested realms allows you to create backbone VPN separation for signaling and media. This means that you can put signaling and media on separate network interfaces, that the signaling and media VPN can have different address spaces, and that the parent realm has one media-only sub-realm.

The following figure shows the network architecture.



In addition, you can achieve enhanced scalability by using a shared service interface. A single service address is shared across many customers/peers, customer specific policies for bandwidth use and access control are preserved, and you can achieve fine-grained policy control.

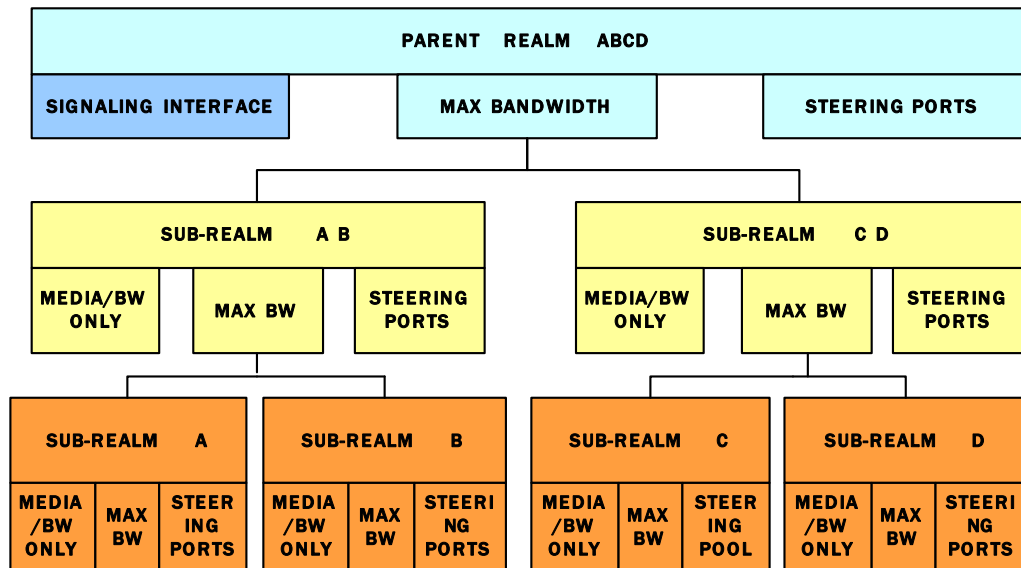
These benefits are achieved when you configure these types of realms:

- **Realm group**—A hierarchical nesting of realms identified by the name of the highest order realm.
- **Controlling realm**—A realms for which a signaling interface is configured. For example, you might configure these signaling interfaces in the following configurations: SIP-NAT, SIP port, H.323 stack, or MGCP. Typically, this is the highest order realm for the parent realm in a realm group.
- **Parent realm**—A realm that has one or more child realms. A parent realm might also be the child realm of another realm group.
- **Child realm**—A realm that is associated with a single higher order parent realm. A child might also be the parent realm of another realm group. Child realms inherit all signaling and steering ports from higher order realms.
- **Media-only realm**—A realm for which there is no configured signaling interface directly associated. Media-only realms are nested within higher order realms.

As these definitions suggest, parent and child realms can be constructed so that there are multiple nesting levels. Lower order realms inherit the traits of the realms above them, including: signaling service interfaces, session translation tables, and steering pools.

Since realms inherit the traits of the realms above them in the hierarchy, you will probably want to map what realms should be parents and children before you start

configuring them. These relationships are constructed through one parameter in the realm configuration that identifies the parent realm for the configuration. If you specify a parent realm, then the realm you are configuring becomes a child realm subject to the configured parameters you have established for that parent. And since parent realms can themselves be children of other realm, it is important that you construct these relationships with care.



Configuring Nested Realms

When you are configuring nested realms, you can separate signaling and media by setting realm parameters in the SIP interface configuration, the H.323 stack configuration, and the steering ports configuration.

- The realm identifier you set in the SIP interface configuration labels the associated realm for signaling.
- The realm identifier you set in the H.323 stack configuration labels the associated realm for signaling.
- The realm identifier you set in the steering ports configuration labels the associated realm for media.

For MGCP, as explained below, you set a special option that enables nested realm use.

Constructing a hierarchy of nested realms requires that you note which realms you want to handle signaling, and which you want to handle media.

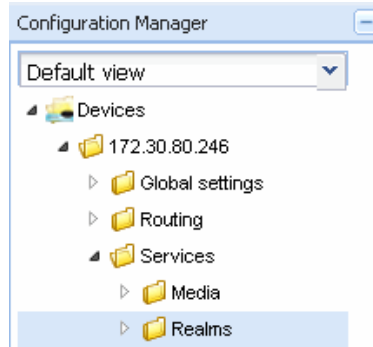
In the SIP port configuration for the SIP interface and in the H.323 stack configuration, you will find an allow anonymous parameter that allows you to set certain access control measures.

Configuring Nested Realms

This section describes how to configure nested realms on your Net-Net SBC from the Net-Net Central, and provides instructions for signaling applications where you are using nested realms.

Choosing the Parent Realm

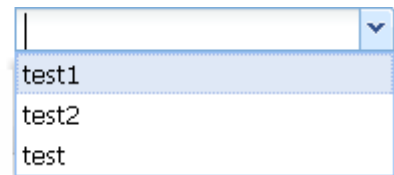
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a Realm and click **Edit**.
4. **Parent realm**—Choose the realm that will be the parent of this realm from the Realm group combination box.

Configuring this parameter makes the realm you are currently configuring the child of the parent you name. As such, the child realm is subject to the configured parameters for the parent.

*Parent realm



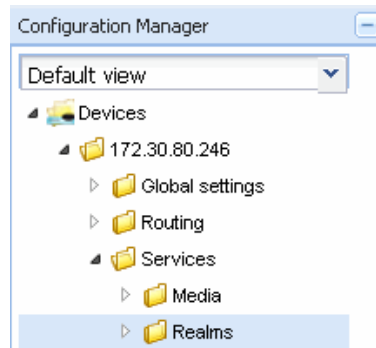
5. Click **Apply**.

Enabling Nested Realms for H.323 Signaling Services

To configure nested realms, you need to set parameters in the realm configuration and in the configurations for H.323 and MGCP signaling protocols.

To configure H.323 stack parameters for nested realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. At the top of the content area, select a Realm and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Click **Add** in the Services and Protocols table.
6. **Service**—Select H323StackConfig from the combination box. The H.323 Stack Configuration now appears in the Services and Protocols table.
7. Select the H.323 realm service and click **Edit**.
8. Click the H.323 service mode tab at the top of the content area.
9. **Allowed connections**—Select all from the combination box to allow all connections.

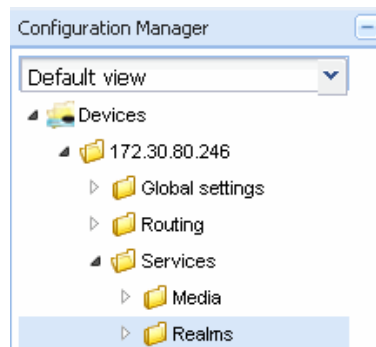


10. Click **Apply**.

Enabling Nested Realms for MGCP Signaling Services

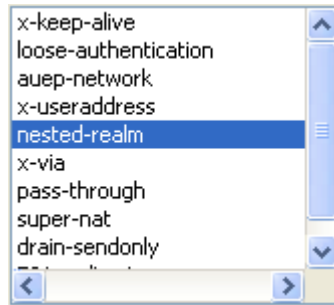
To configure MGCP for nested realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the MGCP realm you want to configure and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.

5. Select the MGCP realm service you want to configure and click **Edit**.
6. Scroll to the Options section, and click **Options**.
The Options dialog box appears.
7. **nested-realm**—Choose the nested-realm option from the Available options list.



8. Click **<>>** to move the nested-realm option to the Selected options list. The option name and description appear in the middle of the screen.
9. Click **OK**.
10. Click **Apply**.

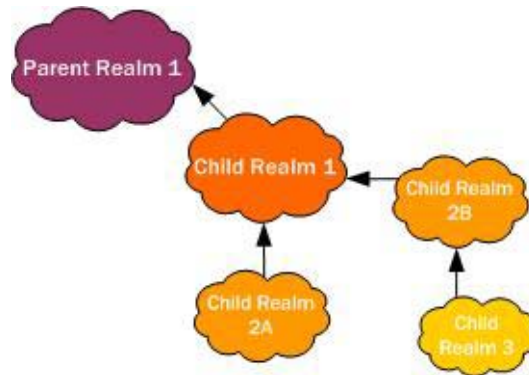
Aggregate Session Constraints: Nested Realms

In addition to setting session constraints per realm for SIP and H.323 sessions, you can also enable the Net-Net SBC to apply session constraints across nested realms. When you set up session constraints for a realm, those constraints apply only to the realm for which they are configured without consideration for its relationship, either as parent or child, to any other realms.

You can also, however, enable the Net-Net SBC to take nested realms into consideration when applying constraints. For example, if a call enters on a realm that has no constraints but its parent does, then the constraints for the parent are applied. This parameter is global and so applies to all realms on the system. For the specific realm the call uses and for all of its parents, the Net-Net SBC increments the counters upon successful completion of an inbound or outbound call.

In the following example, you can see one parent realm and its multiple nested, child realms. Consider applying the following realm constraints:

- Parent Realm 1—55 active sessions
- Child Realm 1—45 active sessions
- Child Realm 2A—30 active sessions
- Child Realm 2B—90 active sessions
- Child Realm 3—20 active sessions



Given the realm constraints outlined above, a call enters the Net-Net SBC on Child Realm 2B, which has an unmet 90-session constraint set. The Net-Net SBC allows the call based on Child Realm 2B. But the call also has to be within the constraints set for Child Realm 1 and Parent Realm 1. If the call fails to fall within the constraints for either of these two realms, then the Net-Net SBC rejects the call.

Impact to Other Session Constraints and Emergency Calls

You can set up session constraints in different places in your Net-Net SBC configuration. Since session agents and SIP interfaces also take session constraints, it is important to remember the order in which the Net-Net SBC applies them:

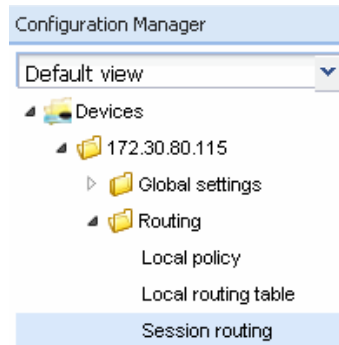
1. Session agent session constraints
2. Realm session constraints (including parent realms)
3. SIP interface session constraints

Emergency and priority calls for each of these is exempt from session constraints. That is, any call coming into the Net-Net SBC marked priority is processed.

Configuring Session Constraints for Nested Realms

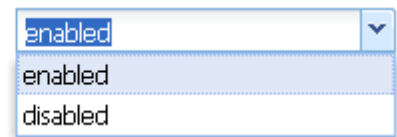
To configure session constraints for nested realms across the entire system:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. **Apply constraints to all related realms**—Select enabled from the combination box to enable session constraints across all nested realms.

Apply constraints to all related realms



5. Click **Apply**.

Aggregate Session Constraints Per Realm

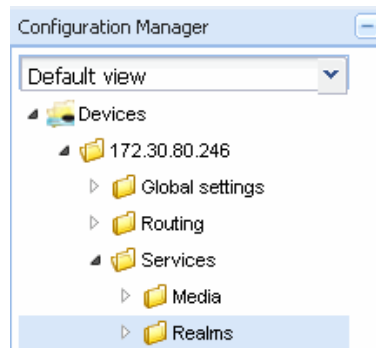
You can set session constraints for the Net-Net SBC's global SIP configuration, specified session agents, and specified SIP interfaces. Setting constraints forces users who have a large group of remote agents to create a large number of session agents and SIP interfaces.

You can also set session constraints for a specific realm. You can group remote agents into one or more realms upon which to apply session constraints. These constraints apply to all SIP and H.323 sessions when enabled.

Configuring Session Constraints per Realm

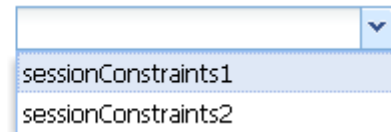
To enable sessions constraints on a per realm basis:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the Realm table and click **Edit**.
4. **Session constraint name**—Choose the name of the constraint you want to use from the drop-down list.

Session constraint name



5. Click **Apply**.

Realm-Based Packet Marking

The Net-Net SBC supports TOS/DiffServ functions that allow you to

- Set up realm-based packet marking by media type, either audio-voice or video
- Set up realm-based packet marking for signaling, either SIP or H.323

Upstream devices use these markings to classify traffic in order to determine the priority level of treatment it will receive.

About TOS/DiffServ

TOS and DiffServ are two different mechanisms used to achieve QoS in enterprise and service provider networks; they are two different ways of marking traffic to indicate its priority to upstream devices in the network.

Given the somewhat confusing differences between TOS and DiffServ (since both specify use of the same byte in the IP header), the ToS byte and DiffServ byte sections below provide some basic information for clarification.

For more information about TOS (packet) marking, refer to:

- IETF RFC 1349 (<http://www.ietf.org/rfc/rfc1349.txt>)

For more information about DiffServ, refer to:

- IETF RFC 2474 (<http://www.ietf.org/rfc/rfc2474.txt>)
- IETF RFC 2475 (<http://www.ietf.org/rfc/rfc2475.txt>).

ToS Byte

The TOS byte format is as follows:

Precedence			TOS				MBZ
0	1	2	3	4	5	6	7

The TOS byte is broken down into three components:

- Precedence—The most used component of the TOS byte, the precedence component is defined by three bits. There are eight possible precedence values ranging from 000 (decimal 0) through 111 (decimal 7). Generally, a precedence value of 000 refers to the lowest priority traffic, and a precedence value of 111 refers to the highest priority traffic.
- TOS—The TOS component is defined by four bits, although these bits are rarely used.
- MBZ—The must be zero (MBZ) component of the TOS byte is never used.

DiffServ Byte

Given that the TOS byte was rarely used, the IETF redefined it and in doing so created the DiffServ byte.

The DiffServ byte format is as follows:

DSCP						CU	
0	1	2	3	4	5	6	7

The DiffServ codepoint value is six bits long, compared to the three-bit-long TOS byte's precedence component. Given the increased bit length, DiffServ codepoints can range from 000000 (decimal 0) to 111111 (decimal 63).

Note: By default, DiffServ codepoint mappings map exactly to the precedence component priorities of the original TOS byte specification.

Packet Marking for Media

You can set the TOS/DiffServ values that define an individual type or class of service for a given realm. In addition, you can specify:

- One or more audio media types for SIP and/or H.323
- One or more video media types for SIP and/or H.323
- Both audio and video media types for SIP and/or H.323

For all incoming SIP and H.323 requests, the media type is determined by negotiation or by preferred codec. SIP media types are determined by the SDP, and H.323 media types are determined by the media specification transmitted during call setup.

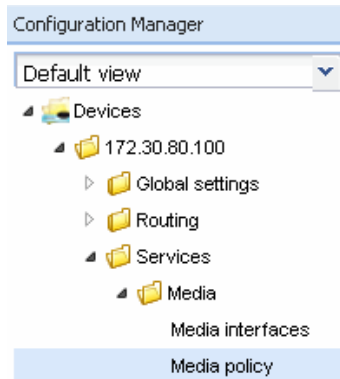
Configuring Packet Marking by Media Type

This section describes how to set up the media policy configuration that you need for this feature, then how to apply it to a realm and how to configure packet marking on your Net-Net SBC from the Net-Net Central. The Net-Net Central system refers to ToS packet marking as QoS marking.

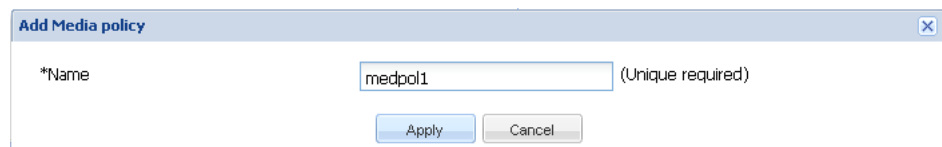
Creating the QoS Marking Profile

Creating the QoS marking profile:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media policy.
3. Click **Add**. The Add Media policy dialog box appears.
4. **Name**—Enter the name of this Media policy; the value you enter uniquely identifies the policy.



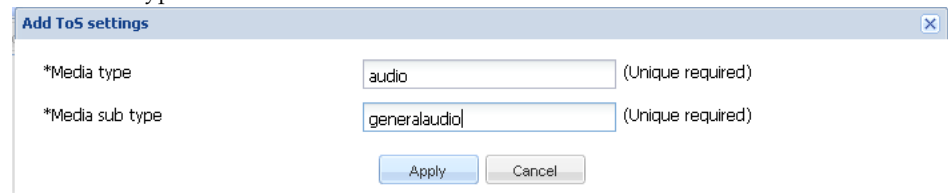
5. Click **Apply**.

The new media policy appears in the table.

6. Select the media policy and click **Edit**.
7. Scroll to the Tos settings table and click **Add**. The Add Tos settings dialog box appears.
8. **Media type**—Select the Media type and value for the profile from the combination box.

Here you can specify one or more audio media types for SIP and/or H.323, one or more video media types for SIP and/or H.323, or both audio and video media types for SIP and/or H.323. The value can be any IANA-defined media type, such as audio, image or model.

9. **Media subtype**—Enter the media sub-type IANA defines for the selected media type.



The 'Add ToS settings' dialog box contains two input fields. The first field, labeled '*Media type', has 'audio' entered and '(Unique required)' to its right. The second field, labeled '*Media sub type', has 'generalaudio' entered and '(Unique required)' to its right. At the bottom are 'Apply' and 'Cancel' buttons.

10. Click **Apply**.
11. Select the Tos configuration from the table and click **Edit**.
12. **ToS value**—Enter the ToS value; this is either a decimal or hexadecimal value.

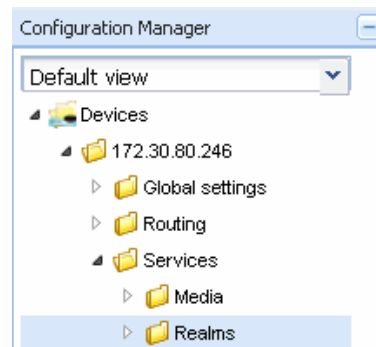
ToS value

13. Click **Apply**.
14. Click **Apply**.

Applying the QoS Marking Profile

To apply the QoS marking profile to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure and click **Edit**.
4. **QoS measurements**—Select enabled from the combination box.

5. **Profile name**—Select the QoS profile name for this realm from the combination box.

QoS

QoS measurements	enabled
Profile name	mp2

6. Click **Apply**.

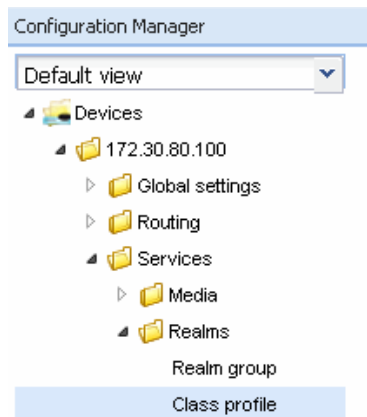
Configuring Packet Marking for Signaling

You must create a QoS profile to support packet marking for signaling. In addition, you need to configure and apply a class policy and class profiles.

Creating a Class Policy

To create a class policy for a class profile:

1. Expand the Configuration Manager slider > Services > Realms.



2. Click Class profile. The Class profile table appears in the content area.
3. Click **Add**. The Add instance dialog box appears.
4. **Profile name**—Enter the name of the class policy; this name value uniquely identifies the class policy.

Add instance

CLASSPOLICY

*Profile name

SIP_marking

(Unique required)

TO

*to

192.168.2.1

*Media policy

mediaPolicy1

Apply

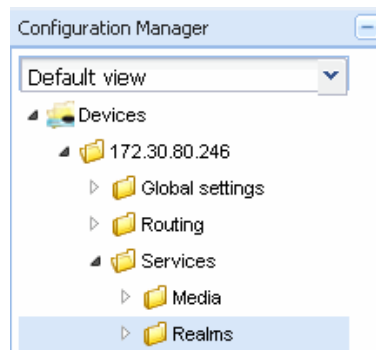
Cancel

5. **To**—Enter the address for the to address in this class policy so that the Net-Net SBC can match incoming traffic for marking. You can use E.164 addresses, a host domain address, or use an asterisk (*) to set all host domain addresses.
6. **Media policy**—Select a media policy from the combination box.
7. Click **Apply** to add this class policy to the class profile. The new policy appears in the Class policy table.
8. You can edit, add, and delete class policies.

Applying a Class Policy to a Realm

To apply a class policy to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to add a class profile to and click **Edit**.
4. Click the Realm tab.
5. **Class profile**—Choose the class policy you want to apply from the combination box.

Class profile

SIP_marking

6. Click **Apply**.

SIP-SDP DCSP Marking/ToS Bit Manipulation

Used to indicate priority and type of requested service to devices in the network, type of service (TOS) information is included as a set of four-bit flags in the IP header. Each bit has a different purpose, and only one bit at a time can be set: There can be no combinations. Available network services are:

- Minimum delay—Used when latency is most important
- Maximum throughput—Used when the volume of transmitted data in any period of time is important
- Maximum reliability—Used when it is important to assure that data arrives at its destination without requiring retransmission
- Minimum cost—Used when it is most important to minimize data transmission costs

The Net-Net SBC's support for type of service (TOS) includes the ability to base classification on the media type as well as the media subtype. In prior releases, you can configure the Net-Net SBC to mark TOS bits on outgoing packets using a media policy. Supported media types include audio, video, application, data, image, text, and message; supported protocol types are H.225, H.245, and SIP. Although H.225 and H.245 are not part of any IANA types, they are special cases (special subtypes) of "message" for the Net-Net SBC. When these criteria are met for an outgoing packet, the Net-Net SBC applies the TOS settings to the IP header. The augmented application of TOS takes matching on media type or protocol and expands it to match on media type, media-sub-type, and media attributes.

The flexibility of this feature resolves issues when, for example, a customer needs to differentiate between TV-phone and video streaming. While both TV-phone and

video streaming have the attribute “media=video,” TV-phone streaming has “direction=sendrcv” prioritized at a high level and video has “direction=sendonly or rcvonly” with middle level priority. The Net-Net SBC can provide the appropriate marking required to differentiate the types of traffic.

How It Works

In the media policy, you can configure values that allow you to create any media type combination allowed by IANA standards. This is a dynamic process because the Net-Net SBC generates matching criteria directly from messages.

The configuration takes a media type value of any of these: audio, example, image, message, model, multipart, text, and video. It also takes a media sub-type of any value specified for the media type by IANA. Using these values, the Net-Net SBC creates a value based on a combination of these values, the Net-Net SBC applies TOS settings.

You also configure the TOS value to be applied, and the media attributes you want to match.

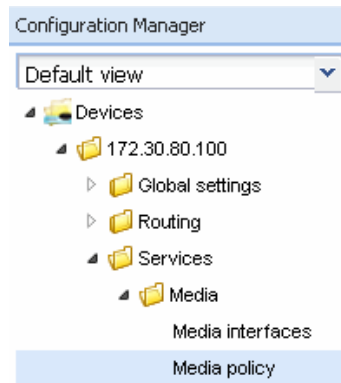
You can have multiple groups of TOS settings for a media policy.

Configuring TOS Bit Manipulation

This section provides instructions for how to configure TOS bit manipulation on your Net-Net SBC.

To configure TOS bit manipulation:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media policy.
3. Select the media policy and click **Edit**.
4. Scroll to the Tos settings table and click **Edit**.
5. In the Media attributes table, click **Add**.
6. **Media attribute**—Enter the media attribute that will match in the SDP. The value is case-sensitive and can be up to 255 characters in length.
7. Click **Apply**.
8. Click **Apply**.

Steering Pools

Steering pools (or realm media addresses) define sets of ports that are used for steering media flows through the Net-Net SBC. These selected ports are used to modify the SDP to cause receiving session agents to direct their media toward this Net-Net system. Media can be sent along the best quality path using these addresses and ports instead of traversing the shortest path or the BGP-4 path.

For example, when the Net-Net SBC is communicating with a SIP device in a specific realm defined by a steering pool, it uses the IP address and port number from the steering pool's range of ports to direct the media. The port the Net-Net SBC chooses to use is identified in the SDP part of the message.

Note: The values entered in the steering pool are used when the Net-Net system provides NAT, PAT, and VLAN translation.

Configuration Overview

To plan steering pool ranges, take into account the total sessions available on the box, determine how many ports these sessions will use per media stream, and assign that number of ports to all of the steering pools on your Net-Net SBC. For example, if your Net-Net SBC can accommodate 500 sessions and each session typically uses 2 ports, you would assign 1000 ports to each steering pool. This strategy provides for a maximum number of ports for potential use, without using extra resources on ports your Net-Net SBC will never use.

The following table lists the steering pool parameters you need to configure:

Parameter	Description
IP address	IP address of the steering pool.
start port	Port number that begins the range of ports available to the steering pool. You must define this port to enable the Net-Net system to perform media steering and NATing.
end port	Port number that ends the range of ports available to the steering pool. You must define this port to enable the Net-Net system to perform media steering and NATing.

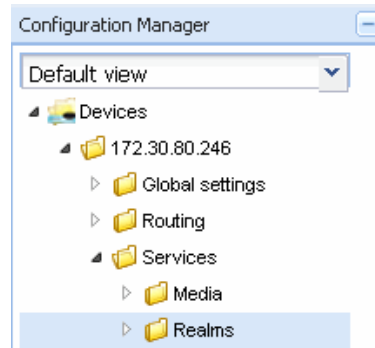
Note: The combination of entries for IP address and start port must be unique in each steering pool. You cannot use the same values for multiple steering pools.

Each bidirectional media stream in a session uses two steering ports, one in each realm (with the exception of audio/video calls that consume four ports). You can configure the start and end port values to provide admission control. If all of the ports in all of the steering pools defined for a given realm are in use, no additional flows/sessions can be established to/from the realm of the steering pool.

Configuring Steering Pools

This section describes how to configure steering pools on your Net-Net SBC from the Net-Net Central. Steering pools are configured within realm objects in the Net-Net Central. Therefore, you do not need to explicitly set the realm when creating a steering pool.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to apply a steering pool to and click **Edit**.
4. Click the Media interface tab.
5. Click **Add**. The Add Realm media address window appears.
6. **IP Address**—Enter the target IP address of the steering pool.
7. **Start port**—Enter the start port of the steering pool; this is the value that begins the range of ports available to this steering pool. The minimum value for this parameter is 0 (default), and the maximum is 65535.

You must enter a valid port number or the steering pool will not function properly.

8. **End port**—Enter the end port of the steering pool; this is the value that ends the range of ports available to this steering pool. The minimum value for this parameter is 0 (default), and the maximum is 65535.

You must enter a valid port number or the steering pool will not function properly.

Add Realm media address
✕

*IP Address	192.168.20.100	(Unique required)
*Start port	40000	(Unique required. Range: 1025..65535, ..0..0. Default: 0)
End port	0	(Range: 1025..65535, ..0..0. Default: 0)
*Realm Id	peer1	(Unique required)

9. Click **Apply**. Your entry will appear in the Realm media address section.
10. Click **Apply**.

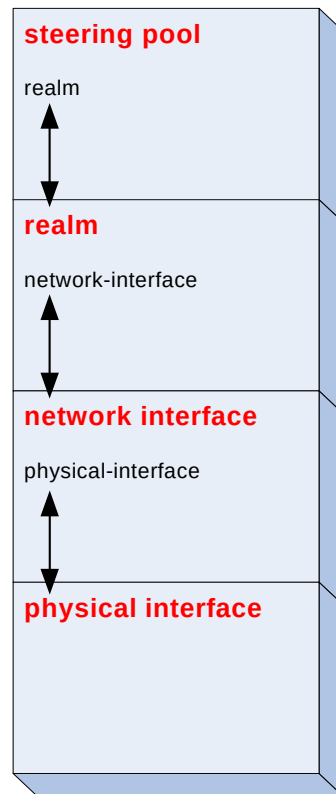
Multiple Interface Realms

The multi-interface realm feature lets you group multiple network interfaces to aggregate their bandwidth for media flows. In effect, this feature lets you use the total throughput of the available physical interfaces on your Net-Net SBC for a single realm. Multi-interface realms are implemented by creating multiple steering pools, each on an individual network interface, that all reference a single realm.

Of course, you can opt not to use this feature and configure your Net-Net SBC to create a standard one-realm to one-network interface configuration.

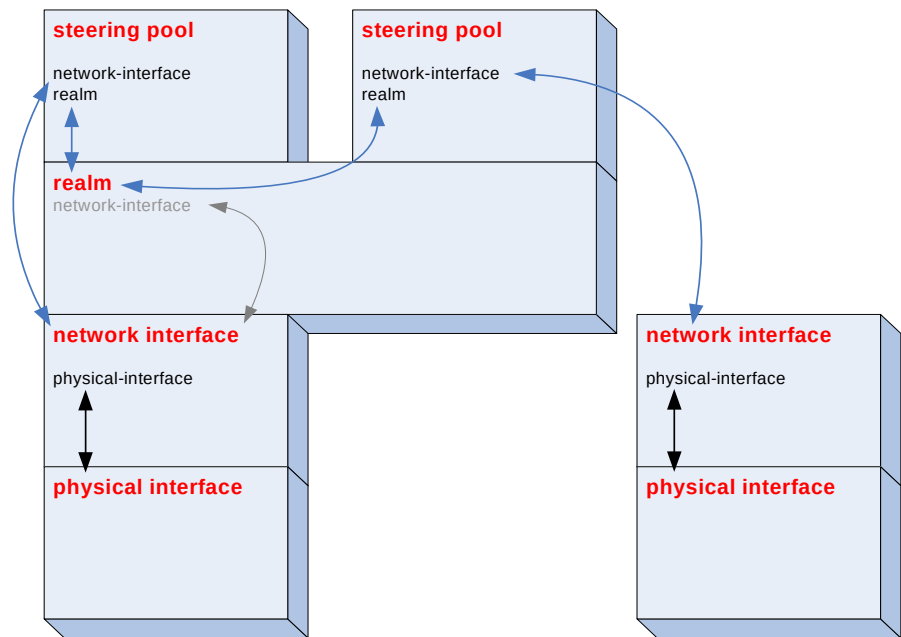
How It Works

Without using multiple interface realms, the basic hierarchical configuration of the Net-Net SBC from the physical interface through the media steering pool looks like this:

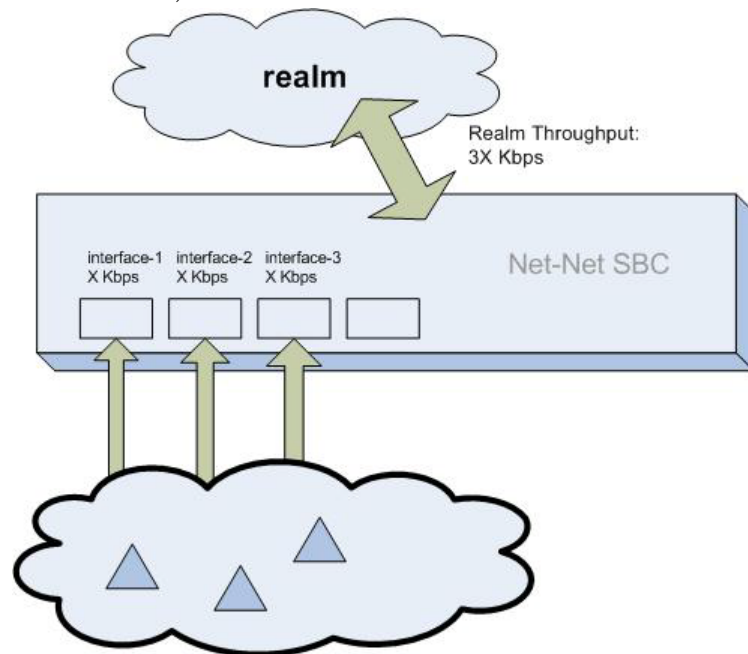


In this model, one (non-channelized) network interface exists on a physical interface. One realm exists on one network interface. One or more steering pools can exist on one realm. Within each higher level configuration element exists a parameter that references a lower level configuration element in the Net-Net SBC's logical network model.

The multi-interface realm feature directs media traffic entering and exiting multiple network interfaces in and out of a single realm. Since all the steering pools belong to the same realm, their assigned network interfaces all feed into the same realm as well. The following diagram shows the relationship in the new logical model:



The advantage of using multi-interface realms is the ability to aggregate the bandwidth available to multiple network interfaces for a larger-than-previously-available total bandwidth for a realm. In the illustration below, three physical interfaces each have X Kbps of bandwidth. The total bandwidth available to the realm with multiple network interfaces is now 3X the bandwidth. (In practical usage, interface-1 only contributes X - VoIP Signaling to the total media bandwidth available into the realm.)



Steering Pool Port Allocation

Every steering pool you create includes its own range of ports for media flows. The total number of ports in all the steering pools that feed into one realm are available for calls in and out of the realm.

Steering pool ports for a given realm are assigned to media flows sequentially. When the first call enters the Net-Net SBC after start-up, it is assigned the first ports on the first steering pool that you configured. New calls are assigned to ports sequentially in the first steering pool. When all ports from the first steering pool are exhausted, the Net-Net SBC uses ports from the next configured steering pool. This continues until the last port on the last configured steering pool is used.

After the final port is used for the first time, the next port chosen is the one first returned as empty from the full list of ports in all the steering pools. As media flows are terminated, the ports they used are returned to the realm's full steering pool. In this way, after initially exhausting all ports, the realm takes new, returned, ports from the pool in a "least last used" manner.

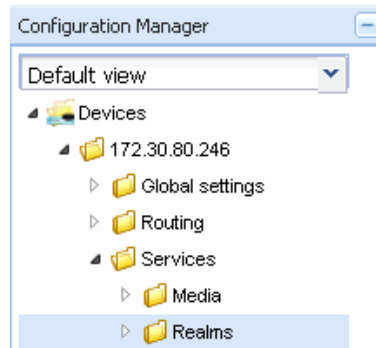
When a call enters the Net-Net SBC, the signaling application allocates a port from all of the eligible steering pools that will be used for the call. Once a port is chosen, the Net-Net SBC checks if the steering pool that the port is from has a defined network interface. If it does, the call is set up on the corresponding network interface. If a network interface is not defined for that steering pool, the network interface defined for the realm is used.

Configuring Multiple Interface Realms

You must first configure multiple physical interfaces and multiple network interfaces on your Net-Net SBC.

To configure steering pools for multi-interface realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm to which you are applying steering pools and click **Edit**.
4. Scroll to Realm media address.
5. Click **Add**. The Realm media address window appears.
6. **IP address**—Enter the IP address of the first steering pool on the first network interface. This IP address must exist on the network interface you use for this configuration
7. **Start port**—Enter the start port of the steering pool; this is the beginning port number of the range of ports for this steering pool.

8. **End port**—Enter the end port of the steering pool; this is the ending port number of the range of ports for this steering pool..

Add Realm media address

*IP Address	192.168.20.100	(Unique required)
*Start port	40000	(Unique required. Range: 1025..65535,..0..0. Default: 0)
End port	0	(Range: 1025..65535,..0..0. Default: 0)
*Realm Id	peer1	(Unique required)

9. Click **Apply**.
10. Repeat this process to configure additional steering pools that use the same network interface.

The properties for each steering pool you create are listed in the Realm media address table. You can add, edit, and delete entries in the table.

11. Click **Apply**.

Media over TCP

The Net-Net SBC now supports RFC 4145 (TCP-Based Media Transport in the SDP), also called TCP Bearer support. Media over TCP can be used to support applications that use TCP for bearer path transport.

RFC 4145 adds two new attributes, *setup* and *connection*, to SDP messages. The *setup* attribute indicates which end of the TCP connection should initiate the connection. The *connection* attribute indicates whether an existing TCP connection should be used or if a new TCP connection should be setup during re-negotiation. RFC 4145 follows the offer/answer model specified in RFC3264. An example of the SDP offer message from the end point 192.0.2.2 as per RFC4145 is as given below:

```
m=image 54111 TCP t38
c=IN IP4 192.0.2.2
a=setup:passive
a=connection:new
```

This offer message indicates the availability of t38 fax session at port 54111 which runs over TCP. Net-Net SBC does not take an active part in the application-layer communication between each endpoint.

The Net-Net SBC provides the means to set up the end-to-end TCP flow by creating the TCP/IP path based on the information learned in the SDP offer/answer process.

TCP Bearer Conditions

The following conditions are applicable to the Net-Net SBC's support of RFC 4145.

1. The Net-Net SBC can not provide media-over-TCP for HNT scenarios (endpoints behind NATs).
2. If media is released into the network, the TCP packets do not traverse the Net-Net. Therefore, no TCP bearer connection is created.
3. The Net-Net SBC does not inspect the *setup* and *connection* attributes in the SDP message since the TCP packets transparently pass through the Net-Net SBC. These SDP attributes are forwarded to the other endpoint. It is the other endpoint's responsibility to act accordingly.
4. After the Net-Net SBC receives a SYN packet, it acts as a pure pass through for that TCP connection and ignores all further TCP handshake messages including FIN and RST. The flow will only be torn down in the following instances:
 - The expiration of one of the following timers (in seconds) you can configure: maximum time before first packet, maximum time between packets, maximum flow time.
 - The whole SIP session is torn down.

TCP Port Selection

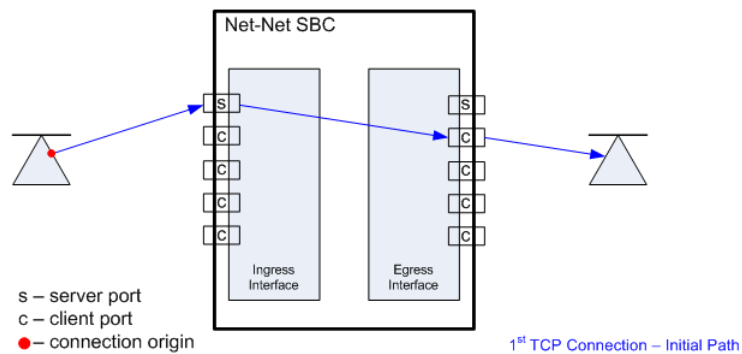
When a call is first set up, the Net-Net SBC inspects the SDP message's m-line to see if any media will be transported via TCP. If the SDP message indicates that some content will use TCP, the Net-Net SBC allocates a configured number of steering ports for the media-over-TCP traffic. These TCP media ports are taken from the each realm's steering pool.

Each endpoint can initiate up to four end-to-end TCP flows between itself and the other endpoint. The Net-Net SBC assigns one port to receive the initial TCP packet (server port), and one to four ports assigned to send TCP traffic (client ports) to the receiving side of the TCP flow. The number of TCP flows for each call is configured globally.

In order to configure the Net-Net SBC to facilitate and support this process, you need to specify the number of ports per side of the call that can transport discrete TCP flows. You can configure one to four ports/flows. For configuration purposes, the Net-Net SBC counts this number as inclusive of the server port. Therefore if you want the Net-Net SBC to provide a maximum of one end-to-end TCP flow, you have to configure two TCP ports; one to receive, and one to send. The receiving port (server) is reused to set up every flow, but the sending port (client) is discrete per flow. For example: for 2 flows in each direction, set the configuration to 3 TCP ports per flow; for 3 flows in each direction, set the configuration to 4 TCP ports per flow, etc.

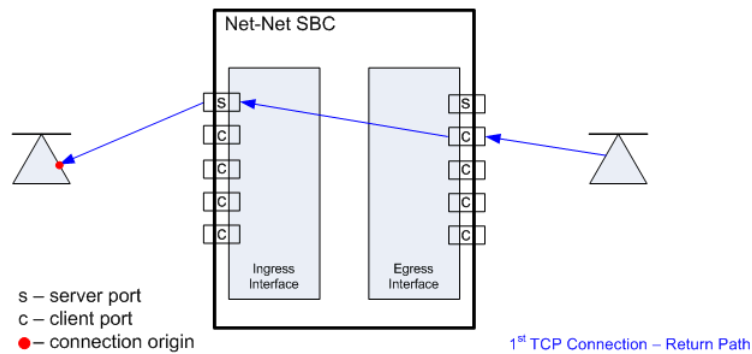
The server port is used for initiating a new TCP connection. An endpoint sends the first packet to a server port on the ingress interface. The packet is forwarded out of the interface through a client port on the egress interface toward an endpoint:

TCP Connection 1 - Eastward Path



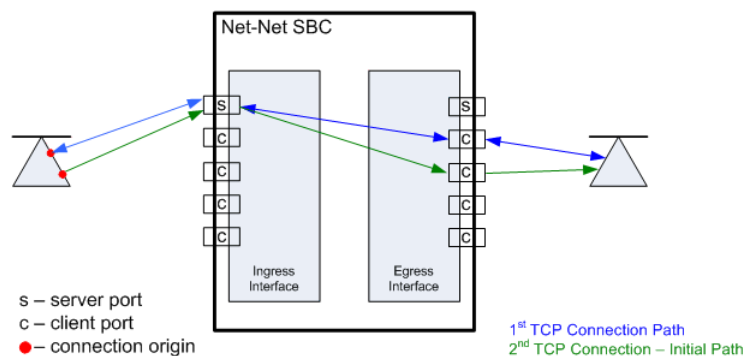
The endpoint responds back to the client port on the egress interface. This message traverses the Net-Net SBC and is forwarded out of the server port on the ingress interface where the initial packet was sent. The remainder of the TCP flow uses the server and client port pair as a tunnel through the Net-Net SBC:

TCP Connection 1 - Westward Path



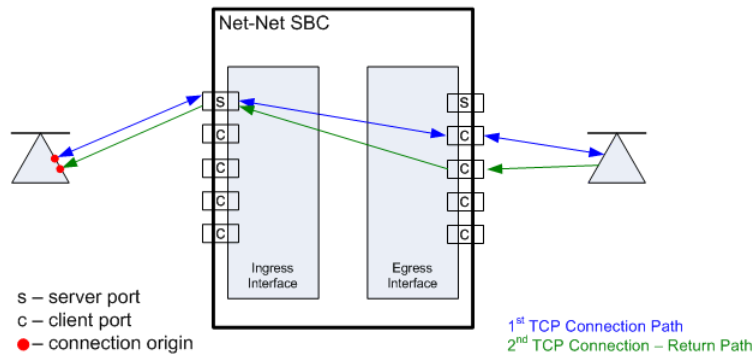
When the second TCP connection is set up in the same direction as in the first example, the first packet is still received on the server port of the ingress interface. The next unused client port is chosen for the packet to exit the Net-Net SBC:

TCP Connection 2 - Eastward Path



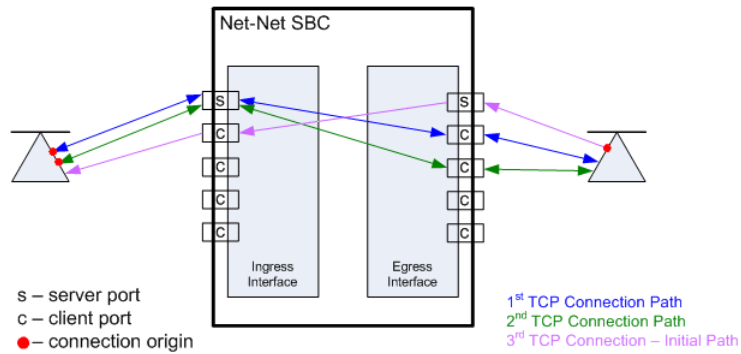
The response takes the same path back to the caller. The remainder of the second TCP connection uses this established path:

TCP Connection 2 - Westward Path



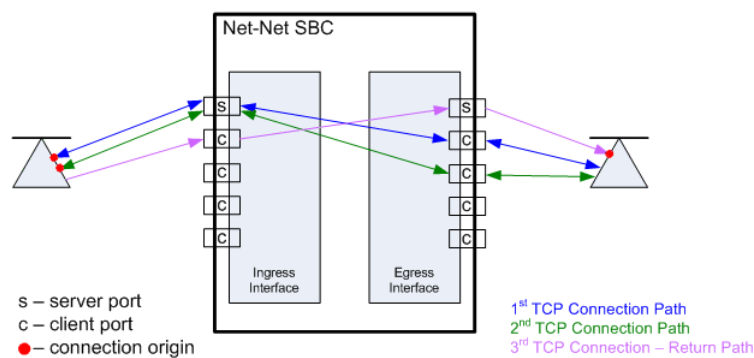
When the callee initiates a TCP connection, it must send its initial traffic to the server port on its Net-Net SBC ingress interface. The packet is forwarded out of the first free client port on the egress side of this TCP connection toward the caller.

TCP Connection 3 – Callee Initiates Connection



The caller's response takes the same path back to the callee that initiated this TCP connection. The remainder of the third TCP connection uses this established path.

TCP Connection 3 – Return Path: Caller to Callee

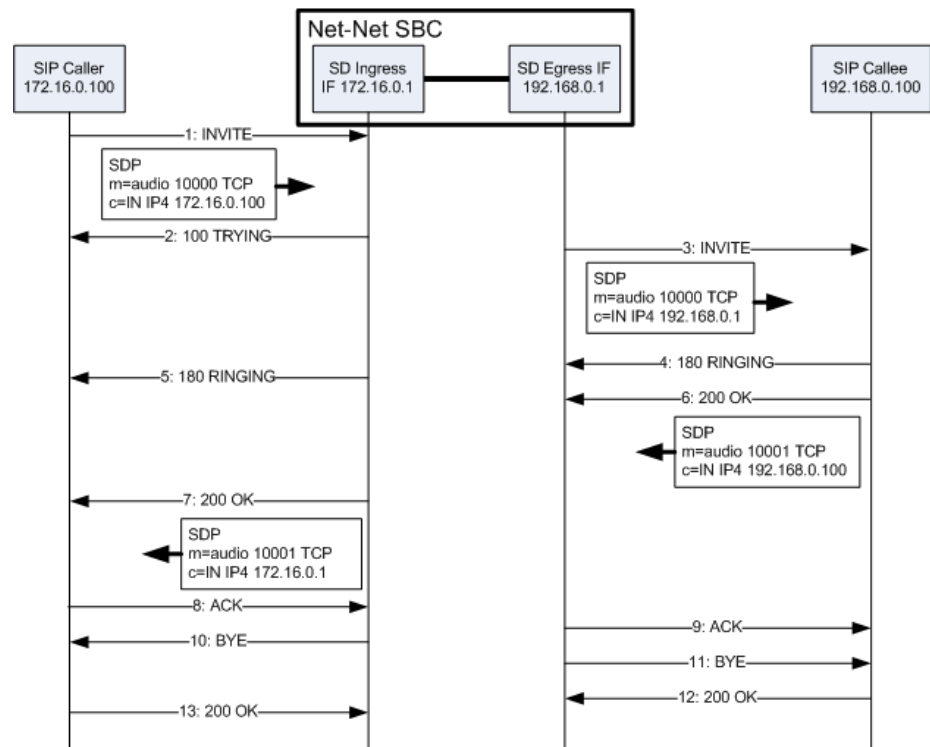


The Net-Net SBC can support a total of eight media-over-TCP connections per call. A maximum of 4 connections are supported as initiated from each side of the call.

SDP Offer Example

The following abbreviated call flow diagram sets up a media-over-TCP flow. Observe that the caller listens for audio over TCP on 172.16.0.10:10000, as described in the SDP offer (1). The Net-Net SBC re-writes the m and c lines in the SDP offer to reflect that it is listening for audio over TCP on its egress interface at 192.168.0.1:10000 (3). The Net-Net SBC then forwards the SIP invite to the callee.

The SIP callee responds with an SDP answer in a 200 OK message. The callee indicates it is listening for the audio over TCP media on 192.168.0.10:10001 (6). The Net-Net SBC re-writes the m and c lines in the SDP answer to reflect that it is listening for audio over TCP on the call's ingress interface at 172.16.0.1:10001 (7). The Net-Net SBC then forwards the SIP invite to the caller.



All interfaces involved with the end-to-end TCP flow have now established their listening IP address and port pairs.

Timers

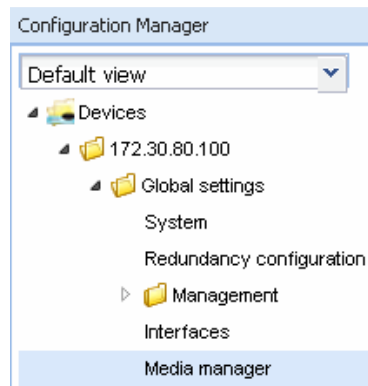
The Net-Net SBC has three guard timers that ensure a TCP media flow does not remain connected longer than configured. You can set each of these from 0 (disabled) to 999999999 in seconds.

- Maximum time before first packet—Sets the maximum time in seconds allowed to elapse between the initial SYN packet and the next packet in this flow.
- Maximum time between packets—Sets the maximum time in seconds allowed to elapse between all subsequent sequential TCP packets.
- Maximum time for flow—Sets the maximum time that a single TCP flow can last. This does not refer to the entire call.

Configuring Media Over TCP

To configure media over TCP (also known as TCP bearer support):

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to TCP bearer support.
5. **Maximum time before first packet (sec)**—Enter the maximum time in seconds allowable before the first packet arrives.
6. **Maximum time between packets (sec)**—Enter the maximum time in seconds allowable between packets.
7. **Maximum time for flow (sec)**—Enter the maximum time in seconds allowable for the TCP flow.
8. **Ports per flow**—Enter the number of ports, inclusive of the server port, to use for media over TCP. The total number of supported flows is this value minus one.

TCP bearer support

Maximum time before first packet (sec)	300
Maximum time between packets (sec)	300
Maximum time for flow (sec)	86400
Ports per flow	2

9. Click **Apply**.

Configuring a Preferred Codec

Certain carriers deploy voice services where their peering partners do not use the carriers' preferred codecs. The Net-Net SBC can now reorder the codecs so that the preferred one is selected first.

Take the example of a carrier that deploys a voice service using G.729 rather than G.711. If that carrier has a peering partner providing call origination for the VoIP customers with G.711 used as the preferred codec, there can be issues with codec selection.

The Net-Net SBC can reorder the default codec in an SDP offer to the preferred codec before it forwards the offer to the target endpoint. When you enable this feature, you increase the probability that the target endpoint will choose the preferred codec for its SDP answer, thereby avoiding use of the undesired codec.

How It Works

You enable the codec reordering feature by setting the `preferred-codec=x` (where X is the preferred codec) option in the realm and session agent configurations. You set it in the realm from which the Net-Net SBC receives SDP offers (in requests or responses), and for which the media format list needs to be reordered by the Net-Net SBC prior to being forwarded. To configure additional codec ordering support for cases when a response or request with an SDP offer is from a session agent, you can set this option in the session agent configuration.

If you enable the option, the Net-Net SBC examines each SDP media description before it forwards an SDP offer. And if necessary, it performs reordering of the media format list to make the preferred codec first in the list of the SDP `m=` line.

The Net-Net SBC determines preferred codecs in the following ways:

- If the response or request with an SDP offer is from a session agent, the Net-Net SBC determines the preferred codec by referring to the session agent configuration. You set the preferred codec for a session agent by configuring it with the `preferred-codec=x` option.
- If the response or request with an SDP offer is not from a session agent or is from a session agent that does not have the `preferred-codec=x` option configured, the Net-Net SBC determines the preferred codec by referring to the `preferred-codec=x` option in the realm.
- If the Net-Net SBC cannot determine a preferred codec, it does not perform codec reordering.

The way that the Net-Net SBC performs codec reordering is to search for the preferred codec in the SDP offer's media description (`m=`) line, and designate it as the codec of choice (if it is not already so). After it marks the preferred codec as the default, the Net-Net SBC does not perform any operation on the remaining codecs in the media format list. Note that the Net-Net SBC performs codec reordering on the media format list only. If the `rtptime` attribute of the preferred codec is present, the Net-Net SBC does not reorder it.

Preferred Codec Precedence

When you configure preferred codecs in session agents and realms, be aware that the codec you set for a session agent takes precedence over one you set for a realm. This means that if you set preferred codecs in both configurations, the one you set for the session agent will be used.

In the case where the Net-Net SBC does not find the session agent's preferred codec in the SDP offer's media format list, then it does not perform codec reordering even if the media format list contains the realm's preferred codec.

Configuring a Preferred Codec

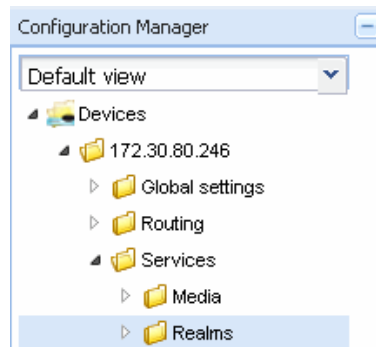
When you configure a preferred codec, the codec you set must match the name of a media profile configuration. If your configuration does not use media profiles, then the name of the preferred codec that you set must be one of the following:

- PCMU
- G726-32
- G723
- PCMA
- G722
- G728
- G729

Note: If you configure this feature for a session agent, you must configure it for the associated realm as well. Otherwise, the feature will not work correctly.

To set a preferred codec for a realm configuration:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm and click edit.
4. Scroll to Options and click **Options**.
5. **preferred-codec**—Choose the preferred-codec option from the Available options list.
6. Click  to move the preferred-codec option to the Selected options list. The option name and description appear in the middle of the screen.

7. **preferred-codec**—In the Option field, set the preferred-codec option to the media profile you want preferred. If you do not have a customized media profile, set to one of the codecs.

Options selector

Available options

- refer-reinvite
- release-media-at-same-nat
- sip-connect-pbx-reg
- preferred-codec**
- li-enabled
- number-normalization

Selected options

- preferred-codec

Option

preferred-codec=PCMU

Description

Set to a string that matches an existing media profile name. If no media profiles are configured, use any of the following strings: PCMU, G726-32, G723, PCMA, G722, G728, or G729.

Edit

User defined option Add

OK Cancel

8. Click **OK**.
9. Click **Apply**.

Restricted Media Latching

The restricted media latching feature lets the Net-Net SBC latch only to media from a known source IP address, in order to learn and latch the dynamic UDP port number. The restricting IP address's origin can be either the SDP information or the SIP message's Layer 3 (L3) IP address, depending on the configuration.

About Latching

Latching is when the Net-Net SBC listens for the first RTP packet from any source address/port for the destination address/port of the Net-Net SBC. The destination address/port is allocated dynamically and sent in the SDP. After it receives a RTP packet for that allocated destination address/port, the Net-Net SBC only allows subsequent RTP packets from that same source address/port for that particular Net-Net SBC destination address/port. Latching does not imply that the latched source address/port is used for the destination of the reverse direction RTP packet flow (it does not imply the Net-Net SBC will perform symmetric RTP).

Restricted Latching

The Net-Net SBC restricts latching of RTP/RTCP media for all calls within a realm. It latches to media based on one of the following:

- SDP: the IP address and address range based on the received SDP c= connect address line in the offer and answer.
- Layer 3: the IP address and address range based on the received L3 IP address of the offer or answer. This option is for access registered HNT endpoints. If the L3 IP address is locally known and cached by the Net-Net SBC as the public SIP contact address, that information could be used instead of waiting for a response. The Net-Net SBC might use the L3 IP address restriction method for all calls regardless of whether the endpoint is behind a NAT or not, for the same realms.

Symmetric Latching

A mode where a device's source address/ports for the RTP/RTCP it sends to the Net-Net SBC that are latched, are then used for the destination of RTP/RTCP sent to the device.

How it Works

After allocating the media session in SIP, the Net-Net SBC sets the restriction mode and the restriction mask for the calling side as well as for the called side. It sets the source address and address prefix bits in the flow. It also parses and loads the source flow address into the MIBOCO messages. After receiving the calling SDP, the Net-Net SBC sets the source address (address and address prefix) in the appropriate flow (the flow going from calling side to the called side). After receiving the SDP from the called side, the Net-Net SBC sets the source address in the flow going from the called side to the calling side.

The Net-Net SBC uses either the address provided in the SDP or the layer 3 signaling address for latching. You also configure the Net-Net SBC to enable latching so that when it receives the source flow address, it sets the address and prefix in the NAT flow. When the NAT entry is installed, all the values are set correctly. In addition, sipd sends the information for both the incoming and outgoing flows. After receiving SDP from the called side sipd, the Net-Net SBC sends information for both flows to the MBCD so that the correct NAT entries are installed.

Enabling restricted latching may make the Net-Net SBC wait for a SIP/SDP response before latching, if the answerer is in a restricted latching realm. This is necessary because the Net-Net SBC does not usually know what to restrict latching to until the media endpoint is reached. The only exception could be when the endpoint's contact/IP is cached.

Relationship to Symmetric Latching

The current forced HNT symmetric latching feature lets the Net-Net SBC assume devices are behind NATs, regardless of their signaled IP/SIP/SDP layer addresses. The Net-Net SBC latches on any received RTP destined for the specific IP address/port of the Net-Net SBC for the call, and uses the latched source address/port for the reverse flow destination information.

If both restricted latching and symmetric latching are enabled, the Net-Net SBC only latches if the source matches the restriction, and the reverse flow will only go to the address/port latched to, and thus the reverse flow will only go to an address of the same restriction.

- Symmetric latching is enabled.

If symmetric latching is enabled, the Net-Net SBC sends the media in the opposite direction to the same IP and port, after it latches to the source address of the media packet.

- Symmetric latching is disabled.

If symmetric latching is disabled, the Net-Net SBC only latches the incoming source. The destination of the media in the reverse direction is controlled by the SDP address.

Example 1

A typical example is when the Net-Net SBC performs HNT and non-HNT registration access for endpoints. Possibly the SDP might not be correct, specifically if the device is behind a NAT. Therefore the Net-Net SBC needs to learn the address for which to restrict the media latching, based on the L3 IP address. If the endpoint is not behind a NAT, then the SDP could be used instead if preferred. However, one can make some assumptions that access-type cases will require registration caching, and the cached fixed contact (the public FW address) could be used instead of waiting for any SDP response.

Example 2

Another example is when a VoIP service is provided using symmetric-latching. A B2BUA/proxy sits between HNT endpoints and the Net-Net SBC, and calls do not appear to be behind NATs from the Net-Net SBC's perspective. The Net-Net SBC's primary role, other than securing softswitches and media gateways, is to provide symmetric latching so that HNT media will work from the endpoints.

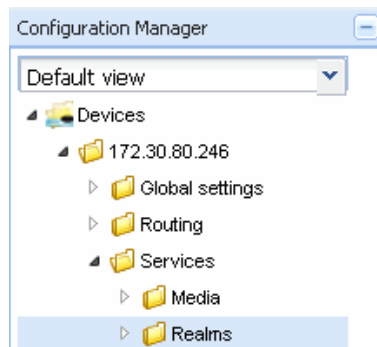
To ensure the Net-Net SBC's latching mechanism is restricted to the media from the endpoints when the SIP Via and Contact headers are the B2BUA/proxy addresses and not the endpoints', the endpoint's real (public) IP address in the SDP of the offer/answer is used. The B2BUA/proxy corrects the c= line of SDP to that of the endpoints' public FW address.

The Net-Net SBC would then restrict the latching to the address in the SDP of the offer from the access realm (for inbound calls) or the SDP answer (for outbound calls).

Configuring Restricted Media Latching

To configure restricted latching:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Scroll to Restricted latching.
4. **Restricted latching mode**—Choose the latching mode from the combination box in the Restricted latching section.
 - **none**—no latching used
 - **sdp**—use the address provided in the SDP for latching
 - **peer-ip**—use the layer 3 signaling address for latching
5. **Length of bit map for sourced latched address**—Enter the number of address bits you want used for the source latched address. If set to 32, the complete IP address is matched. This field will be used only if the restricted latching mode is sdp or peer-ip.

Restricted latching

Restricted latching mode	peer-ip
Length of bit map for source latched address	32

6. Click **Apply**.

Media Release Across SIP Network Interfaces

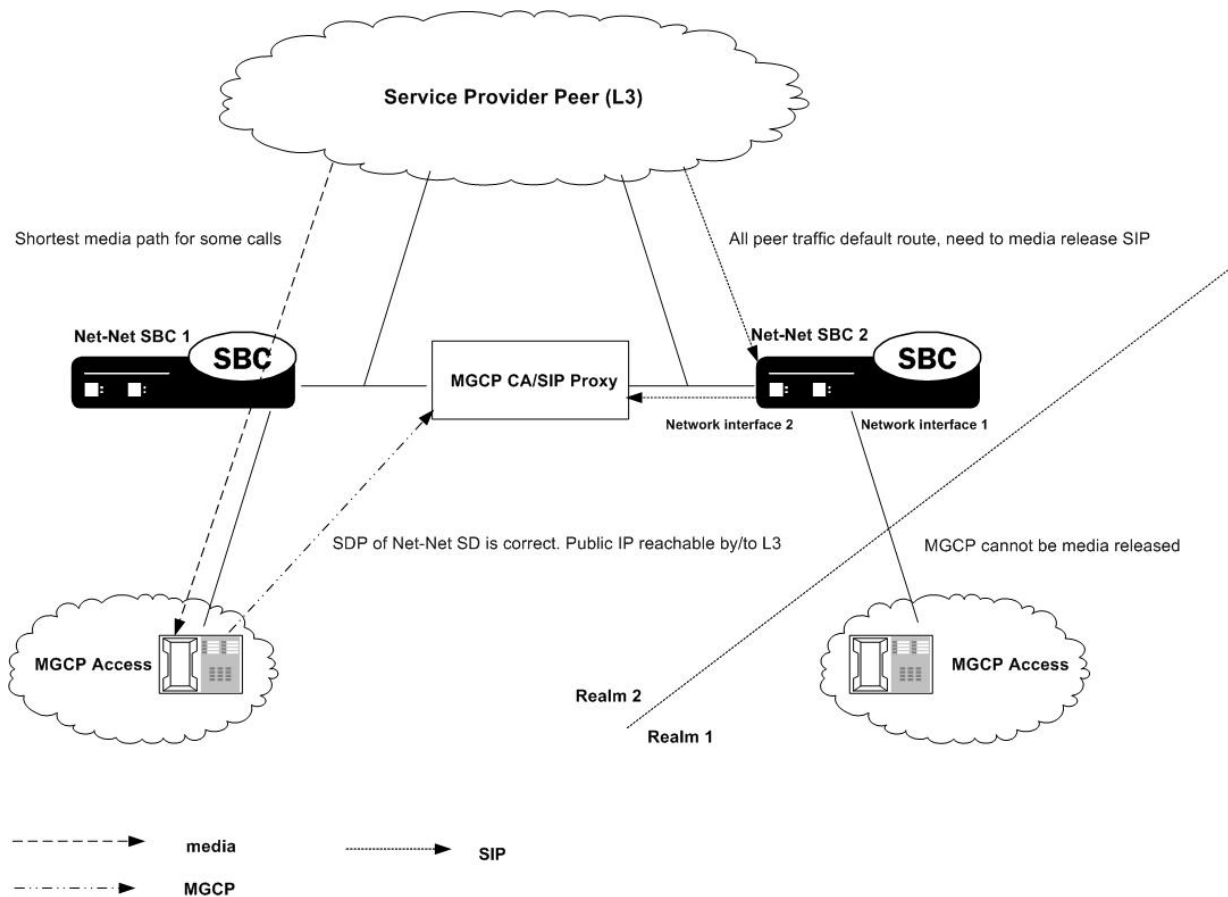
This feature lets the Net-Net SBC release media between two SIP peers, between two realms on two network interfaces of the same Net-Net SBC. Use this feature when you want the Net-Net SBC to release media for specific call flows, regardless of the attached media topology.

Example

You can have two or more Net-Net SBCs with MGCP realms, performing MGCP signaling, media, and NATing to the MGCP call agent. The call agent signals SIP to peers (Level 3) for off-net calls, always through a default Net-Net SBC route. In many cases, the Net-Net SBC being used for SIP call routing (SD2) is not the same Net-Net SBC where the MGCP endpoint resides (SD1). In addition, a more direct media path exists between the MGCP-served Net-Net SBC (SD1) and Level-3. The SDP provided by the Net-Net SBC MGCP ALG (SD1) is public and can be routed to Level 3. However, the SIP default route Net-Net SBC (SD2) is also an MGCP ALG and cannot have global media release. It must keep media management for MGCP.

SIP can also arrive from other Net-Net SBCs (or perhaps go out through them in the future). The Net-Net SBC must be able to perform similar media release for SIP while managing media for MGCP or access SIP realms.

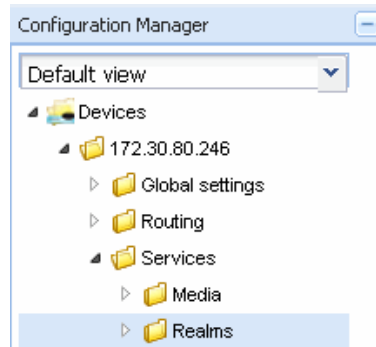
In the following diagram, the access realms for endpoints are currently MGCP, with the expectation they will be migrated to SIP in the future.



Configuring Media Release Across SIP Network Interfaces

To configure media release across network interfaces:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm to which you want to configure media release and click **Edit**.
4. Scroll to Release media across network interfaces.
5. **Release media across network interfaces**—Retain the default value of **enabled** to release media between two SIP peers, between two realms on two network interfaces of the same Net-Net SBC. Set to **disabled** to always release the media, regardless of interface and realm.



6. Click **Apply**.

Media Management Behind the Same IP

The media management behind the same IP feature lets the Net-Net SBC release media when two endpoints are behind the same IP address, in the same realm. You can use this feature for both hosted NAT traversal (HNT) and non-HNT clients.

Additional Media Management Options

Additional media management options include:

- Media directed between sources and destinations within this realm on this specific Net-Net SBC. Media travels through the Net-Net SBC rather than straight between the endpoints.
- Media directed through the Net-Net SBC between endpoints that are in different realms, but share the same subnet.
- For SIP only, media released between multiple Net-Net SBCs.

To enable SIP distributed media release, you must set the appropriate parameter in the realm configuration. You must also set the SIP options parameter to media-release with the appropriate header name and header parameter information. This option defines how the Net-Net SBC encodes IP address and port information for media streams described by, for example, SDP.

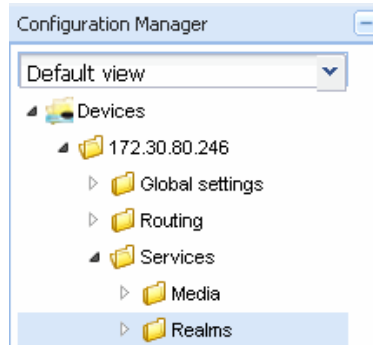
Configuring Media Management Behind the Same IP

You need to configure both the `mm-in-realm` and `mm-same-ip` parameters for the realm:

- If the `Within this realm serviced by this Net-Net system only` is not checked (it is disabled), the `Media release behind same IP` setting is ignored.
- If the `Within this realm serviced by this Net-Net system only` is checked (enabled) and the `Media release behind same IP` setting is disabled, media will be managed in the realm but released if the two endpoints are behind the same IP address.

To configure media management:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm on which you want to configure media management and click **Edit**.
4. Scroll to Manage media between endpoints.
5. **Within this realm serviced by this Net-Net system only**—Select enabled from the combination box.

Manage media between endpoints

Within this realm serviced by this Net-Net system only

enabled
 disabled

Within different realms in same subnet serviced by this Net-Net system only

Within this realm serviced by different Net-Net systems (SIP only)

6. Scroll to the Media release behind same IP parameter.
7. **Media release behind same IP**—Choose enabled from the combination box.

Media release behind same IP

enabled
 disabled

8. Click **Apply**.

Media Release Behind the Same NAT IP Address

You can configure your Net-Net SBC to release media between endpoints sharing the same NAT IP address, even when one of them:

- Is directly addressable at the same IP address as a NAT device, but is not behind a NAT device
- Is at the same IP address of a NAT device the other endpoint is behind

You enable this feature on a per-realm basis by setting an option in the realm configuration.

When this option is not set, the Net-Net SBC will (when configured to do so) release media between two endpoints sharing one NAT IP address in the same realm or network.

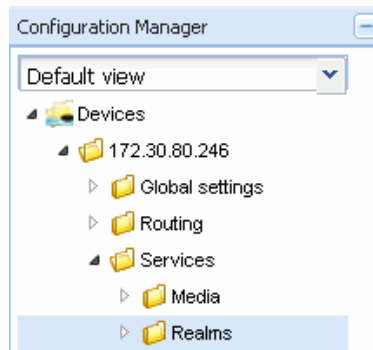
Configuring Media Release Behind the Same NAT IP Address

In order for this feature to work properly, the following conditions apply for the realm configuration:

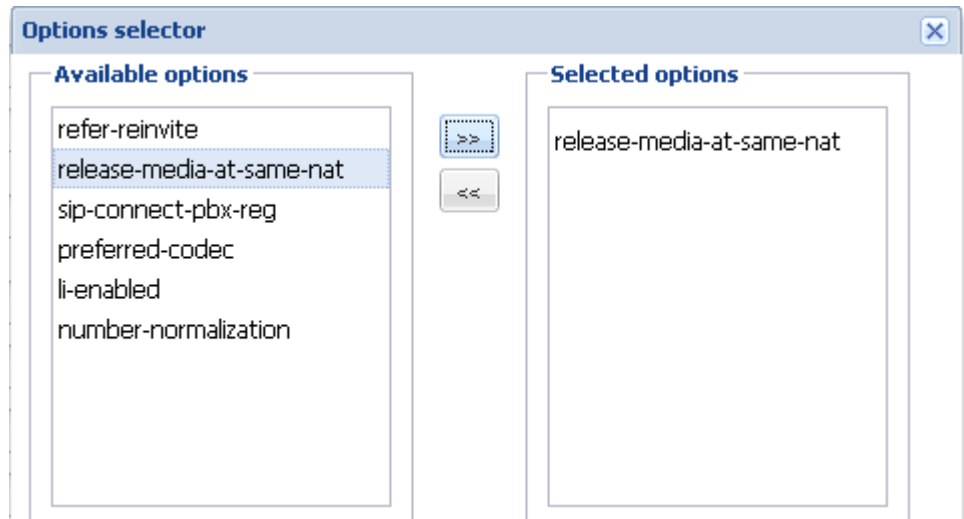
- Either the **Within this realm serviced by this Net-Net system only** or **Within different realms in the same subnet serviced by this Net-Net system only** parameter must be **disabled**; you can have one of these enabled as long as the other is not. The new option will apply to the parameter that is disabled.
- If either **Within this realm serviced by this Net-Net system only** or **Within different realms in the same subnet serviced by this Net-Net system only** parameter is enabled, then the **Media release behind same IP** parameter must be **disabled**.

To enable media release behind the same NAT IP address:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Scroll to Options and click **Options**.
4. Click **release-media-at-same-nat** from the Available options list to select it.
5. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



6. Click **OK**.
7. Click **Apply**.

Per Realm Accounting Control

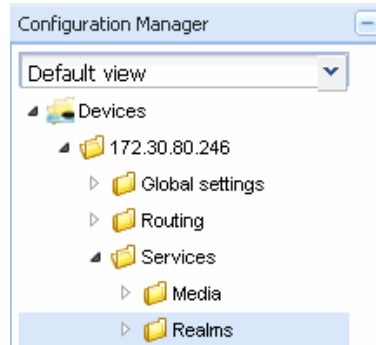
You can enable or disable accounting control for specific realms by setting one parameter. This feature is enabled by default. The Net-Net SBC's SIP and H.323 tasks check whether this parameter is set to enabled or disabled, and sends record on that basis.

For complete details about Remote Authentication Dial-in User Service (RADIUS) accounting, see the *Net-Net 4.1 RADIUS Guide*.

Configuring Per Realm Accounting Control

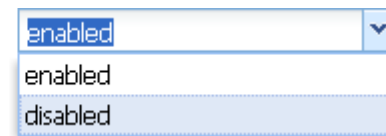
To configure per realm accounting:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm on which you want to configure per realm accounting and click **Edit**.
4. Click the Realm tab.
5. Scroll to Generate CDRs for this realm.
6. **Generate CDRs for this realm**—Select enabled from the combination box to enable the Net-Net SBC to generate CDRs for this realm.

Generate CDRs for this realm



7. Click **Apply**.

SIP Per-User CAC

The Net-Net SBC's call admission control (CAC) has been enhanced to support a greater degree of granularity for SIP sessions. Three new parameters in the realm configuration allow you to define CAC by mode, bandwidth, and sessions.

If you do not enable this feature, the Net-Net SBC performs call admission control (CAC) based on:

- Bandwidth limits configured in realms and nested realms
- Number of media flows available through the steering pool per realm
- Number of inbound sessions configured for a SIP session agent
- Number of total sessions (inbound and outbound) per SIP session agent
- Use of the Net-Net SBC's support for common open policy service (COPS), allowing the Net-Net SBC to perform CAC based on the policies hosted in an external policy server

With these methods, you can configure a basic level of call admission control in order to ensure that a session agent's capacity is not exceeded. You can also ensure that signaling and media bandwidth capacities are not exceeded for physical trunks and peers.

When you enable the SIP per-user CAC feature, the Net-Net SBC only allows the configured number of calls or total bandwidth to and from each user in a particular realm.

How It Works

You should note that the functionality this section describes only works if you enable registration caching on your Net-Net SBC.

For SIP sessions, the Net-Net SBC keeps track of the amount of bandwidth a user consumes and the number of active sessions per address of record (AoR) or per IP address, depending on the CAC mode you select (either `aor` or `ip`). When an endpoint registers with the Net-Net SBC, the Net-Net SBC allots it a total amount of bandwidth and total number of sessions.

Per User CAC Modes

There are three modes that you can set for this feature, and each has an impact on how the other two per-user-CAC parameters are implemented:

- `none`—No per user CAC is performed for users in the realm.
- `aor`—The Net-Net SBC performs per user CAC according to the AoR and the contact associated with that AoR for users in the realm.
- `ip`—The Net-Net SBC performs per user CAC according to the IP address and all endpoints that are sending REGISTER messages from the IP address for users in the realm.

Per User CAC Sessions

You can set the number of CAC for sessions per user in the realm configuration. Depending on the CAC mode you set, the sessions are shared between contacts for the same AoR or the endpoints behind the same IP address.

When it receives an INVITE, the Net-Net SBC determines the registration entry for the calling endpoint and the registration for the called endpoint. It then decides if session can be established between the two. If it can, the Net-Net SBC establishes

the session and changes the active session count for the calling and called endpoints. The count is returned to its original value once the session is terminated.

Per User CAC Bandwidth

You can set the per user CAC bandwidth in realm configuration, too, and it is handled much the same way that the sessions are handled. That is, depending on the CAC mode you set, the bandwidth is shared between contacts for the AoR or the endpoints behind the same IP address. All endpoints must be registered with the Net-Net SBC.

When it receives a Request with SDP, the Net-Net SBC checks to see if there is enough bandwidth for the calling endpoint and for the called endpoint. The Net-Net SBC assumes that the bandwidth usage is symmetric, and it uses the maximum bandwidth configured for the codec that it finds in the Request. In the event that there are multiple streams, the Net-Net SBC determines the total bandwidth required for all of the streams. If the required bandwidth exceeds what is available for either endpoint, the Net-Net SBC rejects the call (with a 503 error response). If the amount of available bandwidth is sufficient, then the used bandwidth value is increased for both the registered endpoints: calling and called. Any mid-session requests for changes in bandwidth, such as those caused by modifications in codec use, are handled the same way.

The Net-Net SBC also keeps track of the bandwidth usage on a global level. When the call terminates, the bandwidth it was consuming is returned to the pool of available bandwidth.

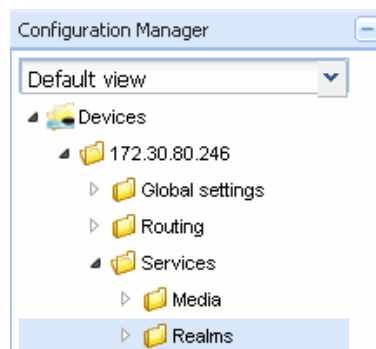
Notes on HA Nodes

This feature has been implemented so that a newly active system is able to perform SIP per user CAC. The standby Net-Net SBC is updated with the appropriate parameters as part of the SIP session update.

Configuring SIP Per-User CAC

To configure SIP per-user CAC on your Net-Net SBC:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm on which you want to configure per realm accounting and click **Edit**.
4. Scroll to Call Admission Control (CAC).
5. **Endpoint identifier**—Select the user CAC mode that you want to use in this realm from the combination box, either `aor` or `ip`. If you do not enter either

of these modes, the Net-Net SBC will not perform per-user CAC; it will remain set to **none** (default).

6. **Maximum bandwidth allowed**—Enter the maximum bandwidth per user for dynamic flows to and from the user.

Leaving this parameter set to 0 (default) means that there is unlimited bandwidth, meaning that the per user CAC feature is disabled in terms of the constraint on bandwidth. The value range you can enter is a minimum of 0, and a maximum of 4294967295.

7. **Maximum sessions allowed**—Enter the maximum number of sessions per user for dynamic flows to and from the user. Leaving this parameter set to 0 (default) means that there is unlimited sessions, meaning that the per user CAC feature is disabled in terms of the constraint on sessions. The value range you can enter is a minimum of 0, and a maximum of 4294967295.

Call Admission Control(CAC)

Endpoint identifier	aor
Maximum bandwidth allowed,Bandwidth constraints	10000
Maximum sessions allowed,Session constraints	10000

8. Click **Apply**.

Bandwidth CAC for Media Release

The bandwidth CAC for media release feature adds per-realm configuration that determines whether or not to include inter-realm calls in bandwidth calculations. When you use this feature, the Net-Net SBC's behavior is to count and subtract bandwidth from the used bandwidth for a realm when a call within a single site has its media released. When you do not enable this feature (and the Net-Net SBC's previous behavior), the Net-Net SBC does not subtract the amount of bandwidth.

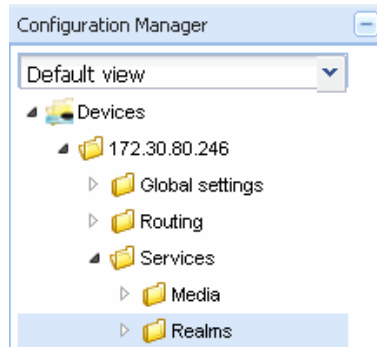
In other words:

- When you enable this feature, an inter-realm media-released call will decrement the maximum bandwidth allowed in that realm with the bandwidth used for that call.
- When you disable this feature (default behavior), and inter-realm media-released call will not decrement the maximum bandwidth allowed for that call.

Configuring Bandwidth CAC for Media Release

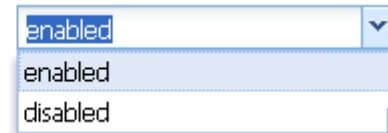
To enable bandwidth CAC for media release:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure in the realm table and click **Edit**.
4. **Include released media in maximum bandwidth allowed**—Select enabled from the combination box to enable this parameter to turn on bandwidth CAC for media release. The default is disabled.

Include released media in maximum bandwidth allowed



5. Click **Apply**.

Media Profiles Per Realm

For different codecs and media types, you can set up customized media profiles that serve the following purposes:

- Police media values
- Define media bandwidth policies
- Support H.323 slow-start to fast-start interworking

You can use media policies globally for the Net-Net SBC, or—starting with Release C6.1.0—you can configure them for application on a per-realm basis. For a realm, you can configure a list of media profiles you want applied. The Net-Net SBC matches the value you set for the **match media profiles** parameter, and then applies those media profiles to the realm itself and to all of its child realms (but not to its parent realms).

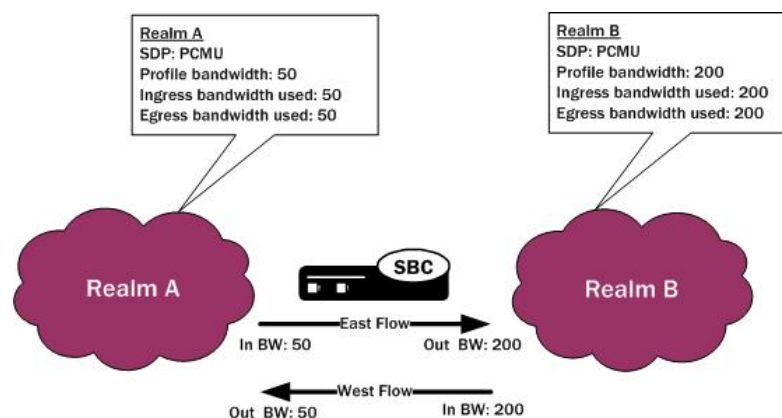
Note: This feature has no impact on the ways the Net-Net SBC uses media profiles non-realm applications such as: H.323 interfaces, SIP interfaces, IWF, session agents, codec policies, and policy attributes.

Call Admission Control and Policing

The Net-Net SBC supports call admission control (CAC) based on realm, and it applies the limits on either ingress or egress bandwidth counters. If a calls exceeds bandwidth on either the ingress or egress side, the Net-Net SBC rejects the call. You

can also use per-user CAC, which limits the maximum bandwidth from the east and west flows for both the TO and FROM users.

When you apply media profiles to a realm, the Net-Net SBC applies bandwidth policing from the flow's ingress realm media profile. In the diagram below, the Net-Net SBC policies traffic for Realm A based Realm A's policing values, and the same is true for Realm B.



Configuring Multiple Media Profiles Per Realm

This section shows you how to configure multiple media profiles per realm, and it explains how to use wildcarding.

Releases C6.1.0 and later accept the subname so you can configure multiple media profile for the same codec; the codec **name** customarily serves and the name value for a media profile configuration.

About Wildcarding

You can wildcard both portions (name and subname) of this value:

- When you wildcard the **name** portion of the value, you can provide a specific subname that the Net-Net SBC uses to find matching media profiles.
- When you wildcard the subname portion of the value, you can provide a specific **name** that the Net-Net SBC uses to find matching media profiles.

You can also enter the name value on its own, or wildcard the entire value. Leaving the subname value empty is also significant in that it allows the realm to use all media profile that have no specified **subname**. However, you cannot leave the **name** portion of the value unspecified (as all media profiles are required to have names).

The following table lists examples of the syntax:

Syntax	Example Value	Description
<name>	PCMU	Matches any and all media profiles with the name value configured as PCMU. This entry has the same meaning as a value with this syntax: <name>::*.
<name>::	PCMU::	Matches a media profile with the name with the name value configured as PCMU with an empty subname parameter.
<name>::<subname>	PCMU::64k	Matches a media profiles with the name with the name value configured as PCMU with the subname parameter set to 64k.

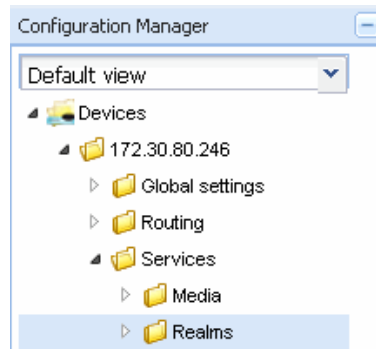
Syntax	Example Value	Description
*	*	Matches anything, but does not have to be a defined media profile.
:: <td>::<td>Matches any and all media profiles, but requires the presence of media profile configurations.</td></td>	*:: <td>Matches any and all media profiles, but requires the presence of media profile configurations.</td>	Matches any and all media profiles, but requires the presence of media profile configurations.
:: <subname><td>::64k</subname><td>	Matches all media profiles with this subname. You might have a group of media profiles with different names, but the same subname value.	
:: <td>::<td>Matches any media profiles with an empty subname parameter.</td></td>	*:: <td>Matches any media profiles with an empty subname parameter.</td>	Matches any media profiles with an empty subname parameter.
::	::	Invalid
::*	::*	Invalid

The Net-Net SBC performs matching for wildcarded **match media profiles** values last. Specific entries are applied first and take precedence. When the Net-Net SBC must decide between media profiles matches, it selects the first match.

Customizing Media Profiles

To use media profiles for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm and click **Edit**.
4. Scroll to Media profile applied to realm.
5. Click **Add**. The Add instance dialog box appears.
6. **Match media profile**—Enter the media profile you would like applied to this realm from the combination box. This value corresponds to the name and subname parameters in the media profile configuration. You can wildcard either of these portions of the value, or you can leave the <subname> portion empty. Refer to the [About Wildcarding](#) section above for more information about syntax, wildcarding, and empty values.

This parameter has no default.

The screenshot shows a window titled "Add instance" with a close button in the top right corner. Inside the window, the text "realmConfig" is displayed. Below it, the configuration path "REALMCONFIG/MATCHMEDIAPROFILE" is shown. A label "*Media profile" is followed by a text input field containing "mediaprof1". To the right of the input field, the text "(Unique required)" is displayed. At the bottom of the window, there are two buttons: "Apply" and "Cancel".

7. Click **Apply**.
8. To add additional media profiles, repeat steps 5 through 5.
9. Click **Apply**.

Multiple Media Profiles

You can use the media profiles configuration to set up:

- One media profile for a particular SIP SDP encoding (such as G729), where the name of the profile identifies it uniquely. This behavior is your only option in Net-Net OS release prior to Release C6.1.0.
- Multiple media profiles for the same SIP SDP encoding. Available in Release C6.1.0 and forward, you can create multiple media profiles for the same encoding. To do so, you add a subname to the configuration, thereby identifying it uniquely using two pieces of information rather than one.

The sections below provide two descriptions of deployments where using multiple profiles for the same codec would solve codec and packetization problems for service providers.

Use Case 1

Service Provider 1 peers with various carriers, each of which uses different packetization rates for the same codec. For example, their Peer 1 uses 10 milliseconds G.711 whereas their Peer 2 uses 30 milliseconds for the same codec. The difference in rates produces a difference in bandwidth consumption—resulting in a difference in SLA agreements and in Net-Net SBC call admission control (CAC) and bandwidth policing. Service Provider 1 uses the Net-Net SBC's media profile configuration parameters to determine CAC and bandwidth policing. Because this service provider's peers either do not use the SDP p-time attribute or use it inconsistently, it is difficult to account for bandwidth use. And so it is likewise difficult to set up meaningful media profiles.

The best solution for this service provider—given its traffic engineering and desire for the cleanest routing and provisioning structures possible—is to define multiple media profiles for the same codec.

Use Case 2

Service Provider 2 supports H.263 video, for which the Net-Net SBC offers a pre-provisioned media profile with a set bandwidth value. And yet, H.263 is not a codec that has a single bandwidth value. Instead, H.263 can have different bandwidth values that correspond to various screen resolution and quality. While it is true that the Net-Net SBC can learn the requisite bandwidth value from SDP, not all SDP

carries the bandwidth value nor do system operators always trust the values communicated.

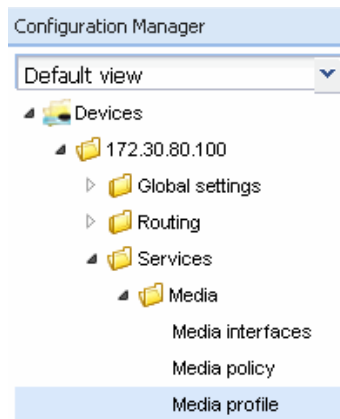
Configuring multiple media profiles for the same codec (here, H.263) helps considerably with this problem—and moves closer to complete solution. Service Provider 2 can configure H.263 media profiles capable of handling the different bandwidth values that might appear.

Configuring a Subname for a Media Profile

Configuring the **subname** parameter in the media profiles configuration allows you to create multiple media profiles with the same name.

To configure a subname for a media profile:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**. The Add instance dialog box appears.
4. **Subname**—Type the subname value for this media profile in the text box. Information such as the rate or bandwidth value make convenient subname values. For example, you might set the **name** of the media profile as PCMU and the **subname** as 64k.

This parameter is not required and has no default.

The 'Add instance' dialog box is shown. It has a title bar with 'Add instance' and a close button. The main area is titled 'MEDIAPROFILE'. It contains two fields: '*Name' with a dropdown menu showing 'PCMU' and '(Unique required)' next to it, and '*Subname' with a text input field containing '64k' and '(Unique required)' next to it. At the bottom, there are 'Apply' and 'Cancel' buttons.

5. Click **Apply**.

STUN Server

The Net-Net SBC supports RFC 3489, which defines Simple Traversal User Datagram Protocol (UDP) through Network Address Translators (NATs). Known as STUN, this is a lightweight protocol that allows applications to:

- Discover the presence and types of both NATs and firewalls between themselves and the public Internet
- Determine the public IP addresses allocated to them by the NAT

SIP endpoints use the STUN protocol to find out the public IP addresses and ports for SIP signaling and RTP media transport. Then they can use the address and port information to create multimedia sessions with other endpoints on the public network.

You can define STUN servers functionality on a per-realm basis, allowing you set up multiple STUN servers.

About STUN Messaging

STUN messages use three Binding messages and three Shared Secret messages. While it supports all three Binding messages (request, response, and error), the Net-Net SBC does not support the Shared Secret Request, or the message integrity mechanism, that relies upon the shared secret. When acting as a STUN server, the Net-Net SBC responds to STUN binding requests in accordance with RFC 3489 and the rfc3489bis draft.

STUN messages can contain the following attributes:

Message Type	Attribute Description
MAPPED-ADDRESS	Appears in the Binding Response; contains the source IP address and port from which the Binding Request was sent to the STUN server.
XOR-MAPPED-ADDRESS	Appears in the Binding Response; contains the MAPPED-ADDRESS information encoded in a way the prevents intelligent NAT devices from modifying it as the response goes through the NAT.
SOURCE-ADDRESS	Appears in the Binding Response; contains the IP address and port from which the STUN server sent its response.
CHANGED-ADDRESS	Appears in the Binding Response; contains an alternate STUN server IP address and port, different from the primary STUN server port. The STUN client might use this attribute to perform the NAT tests described in RFC 3489.
CHANGE-REQUEST	Appears in the Binding Request; instructs the STUN server to send its response from a different IP address and/or port. The STUN client might use this attribute to perform the NAT tests described in RFC 3489.
RESPONSE-ADDRESS	Appears in the Binding Request; defines an IP address and port to which the STUN server should send its responses. Appears in the Binding Request;
REFLECTED-FROM	Appears in the Binding Response; reflects the IP address and port from which a Binding Request came. Only included when the Binding Request has used the RESPONSE-ADDRESS attribute.

Message Type	Attribute Description
UNKNOWN-ATTRIBUTES	Appears in the Binding Error; reflects the mandatory attributes in a Binding Request message that the server does not support.
ERROR-CODE	Appears in the Binding Error; indicates an error was detected in the Binding Request, and contains an error code and reason phrase.

To perform NAT discovery, the endpoint (STUN client) sends a Binding Request to the STUN server port (IP address and port) with which it is configured. The STUN server then returns either a;

- Binding Response—Allows the transaction to proceed
- Binding Error—Halts the transaction, and prompts the client to take the action appropriate to the response given in the ERROR-CODE attribute

When the transaction proceeds and the STUN server sends the Binding Response, that response contains the MAPPED-ADDRESS attribute, which contains the IP address and port from which the server received the request. The STUN client then uses the MAPPED-ADDRESS when sending signaling messages.

For example, a SIP endpoint sends Binding Requests from its SIP port to determine the public address it should place in SIP headers, like the Via and Contact, of the SIP requests it sends. When this SIP endpoint prepares to make or answer a call, it sends Binding Requests from its RTP port to find out the public address it should place in SDP included in an INVITE request or response.

STUN Server Functions on the Net-Net SBC

When the Net-Net SBC receives a STUN message, it first determines its message type. Only STUN Binding Requests are processed, and all other message types are dropped without response.

Then the Net-Net SBC examines the Binding Request's STUN attributes. It returns error responses if it finds any unsupported mandatory attributes. This takes the form of a Binding Error Response, containing the ERROR-CODE attribute with reason 420 (*Unknown Attribute*) and an UNKNOWN-ATTRIBUTES attribute with a list of the unsupported attributes. If the Net-Net SBC receives a Binding Request with attributes that do not belong in STUN Binding Requests, it returns the Binding Error Response with the ERROR-CODE attribute with reason 400 (*Bad Request*).

Next the Net-Net SBC determines whether to follow RFC 3489 procedures or rfc3489bis procedures. If the Transaction ID contains the STUN cookie, then the Net-Net SBC follows rfc3489bis procedures; if not, it follows RFC 3489 procedures. Because it defines the procedures for testing the NAT to see what type of NAT it is, RFC 3489 procedures are most complex. Issues with reliability of those results have caused testing procedures and attributes to be deprecated in rfc3489bis.

RFC 3489 Procedures

The Net-Net SBC (the STUN server) constructs the Binding Response and populates it with these attributes:

- MAPPED-ADDRESS and (optionally) XOR-MAPPED-ADDRESS—Containing the source IP address and port from which the server saw the request come
- SOURCE-ADDRESS—Containing the IP address and port from which the server will send the Binding Response

- **CHANGED-ADDRESS**—Containing the STUN server port that has a different address and different port from the ones on which the server request was received

If the Binding Request contains a **RESPONSE-ADDRESS** attribute, the server adds the **REFLECTED-FROM** attribute with the IP address and port from which the server saw the request come. Then the server sends the Binding Response to the IP address and port in the **RESPONSE-ADDRESS** attribute. If the **RESPONSE-ADDRESS** attribute's IP address and port are invalid, the server sends a Binding Error Response with an **ERROR-CODE** attribute reason **400 (Bad Request)** to the client.

If the Binding Request contains a **CHANGE-REQUEST** attribute, the server sends Binding Response from the IP address and port matching the information in the **CHANGE-REQUEST**. The following variations can occur:

- If the IP address and port flags are set, the server selects the server port with a different IP address and different port.
- If only the IP address flag is set, the server selects the server port with a different IP address but with the same port.
- If only the port flag is set, the server selects the server port with the same IP address but with a different port.

The selected server port appears in the Binding Responses's **SOURCE-ADDRESS** attribute. When there is no **CHANGE-REQUEST** attribute, the server uses the server port on which the Binding Request was received.

Finally, the server encodes the outgoing message and sends it to the client at either:

- The destination IP address and port in the **REPOSE-ADDRESS** attribute, if it was present in the Binding Request.
- The **MAPPED-ADDRESS**.

rfc3489bis Procedures

If the Binding Request contains the appropriate cookie in its Transaction ID, the server constructs a Binding Response populated with the **XOR-MAPPED-ADDRESS** attribute. That attribute will contain the source IP address and port from which the server saw the request come. Then the server encodes and sends the message to the client from the IP address and port on which the request was received. The message is sent to the IP address and port from which the request came.

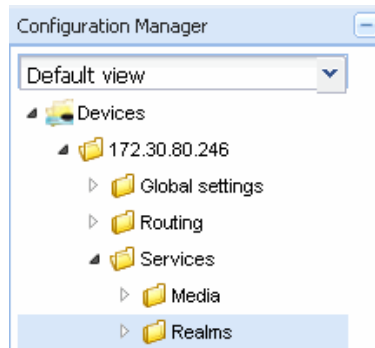
Configuring a STUN Server

You configure STUN servers on a per-realm basis, one server per realm. To support the various NAT tests it describes, RFC 3489 requires that two different IP addresses and two different UDP port numbers be used for each server. Your STUN server will listen on a total of four STUN server ports. Although newer work does away with this requirement, the Net-Net SBC supports it for the purpose of backwards compatibility.

For each realm configuration with an enabled STUN server, untrusted ACL entries will be added to forward all packets received on the four STUN Server Port.

To enable STUN server support for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. Scroll to STUN.
5. **Enable server**—Select enabled from the combination box to activate STUN server support for this realm. This parameter defaults to disabled, meaning STUN server support is off.
6. **Server IP address**—Enter the IP address for the primary STUN server port. The default for this parameter is 0.0.0.0.
7. **Server IP port**—Enter the port number to use with the **Server IP address** for primary STUN server port. The default is 3478.
8. **Changed server IP address**—Enter the IP address for the CHANGED-ADDRESS attribute in Binding Requests received on the primary STUN server port. This IP address must be different from the one defined for the **Server IP address** parameter. The default for this parameter is 0.0.0.0.
9. **Changed server IP port**—Enter the port combination to define the CHANGED-ADDRESS attribute in Binding Requests received on the primary STUN server port. The default for this parameter is 3479.

STUN

Enable server	enabled
Server IP address	0.0.0.0
Server IP port	3478
Changed server IP address	0.0.0.0
Changed server IP port	3479

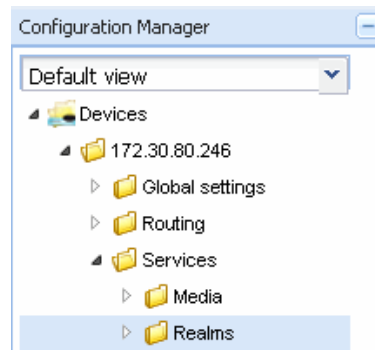
10. Click **Apply**.

Multi-System Selective SRTP Pass-Through

Configuring Multi-System Selective SRTP Pass-Through

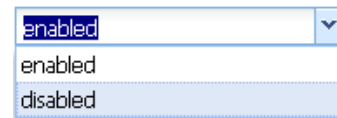
To configure multi-system selective SRTP pass-through:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the Realm table and click **Edit**.
4. Click the Realm tab.
5. Scroll to the bottom of the window.
6. **SRTP MSM passthrough**—Click enabled or disabled in the drop down list to enable this feature. This feature is disabled by default.

SRTP MSM passthrough



7. Click **Apply**. A Success message appears.
8. Click **OK**.

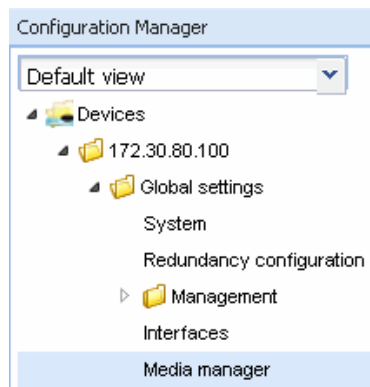
IP Identification (ID) Field

By default, non-fragmented UDP packets generated by media interfaces have the ID field set to 0. You can configure the Net-Net DD to populate this field with an incrementing value by adding the **increment-ip-id** option in the media manager. Every non-fragmented packet sent will have its ID increased by one from the previous packet sent.

Using a packet trace application, egress packets from the SD will have an ID field that appears to be incrementing. Enabling the ID field can help distinguish a retransmitted non-fragmented application layer packet from a packet retransmitted by the network layer in monitoring or lab situations.

To enable ID field generation in media-manager:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to the Options section, and click **Options**. The Options dialogue box appears.
5. Click **increment-ip-id** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

Introduction

This chapter explains how to configure the Net-Net SBC to support Session Initiation Protocol (SIP) signaling services for hosted IP services applications. SIP is a text-based application-layer signaling protocol that creates, identifies, and terminates multimedia sessions between devices.

About the Net-Net SBC and SIP

This section describes the Net-Net SBC's support of SIP. It provides the basic information you need to understand before you configure the Net-Net SBC for SIP signaling.

Types of SIP Devices

There are four types of SIP devices:

- User agent (UA) is an endpoint in SIP end-to-end communication. A UA is a user agent client (UAC) when it initiates a request and waits to receive a response. A UA is a user agent server (UAS) when it receives a request and generates a response. A given UA will be a UAC or a UAS depending on whether it is initiating the request or receiving the request.
- Proxy (or proxy server) is an intermediary entity that acts as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server's primary role is routing. Its job is to ensure that a request is sent to another entity closer to the targeted user. A proxy interprets, and if necessary, rewrites specific parts of a request message before forwarding it.
- Redirect server is a UAS that generates redirect responses to requests it receives, directing the client to contact an alternate set of targets. Unlike a proxy which forwards the request to the alternate set of targets, the redirect response tells the UAC to directly contact the alternate targets.
- Registrar is a server that accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles. Proxies and redirect servers can use the information from the location service to determine the location of the targeted user.

A redirect server and a registrar are each a special type of UA because they act as the UAS for the requests they process.

Basic Service Models

The Net-Net SBC operates as a back-to-back user agent (B2BUA) within the following two basic service models:

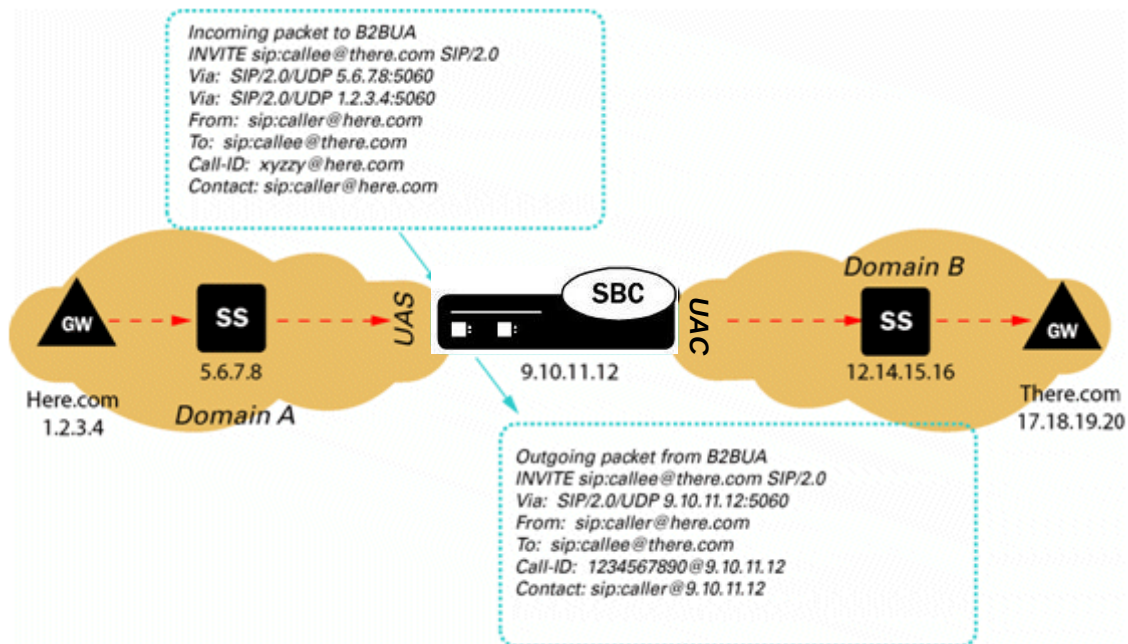
- peering
- hosted IP services

About B2BUA

A B2BUA is a logical entity that receives a request and processes it as a user agent server (UAS). In order to determine how the request should be answered, it acts as a user agent client (UAC) and generates requests. It maintains dialog state and must participate in all requests sent on the dialogs it has established.

SIP B2BUA Peering

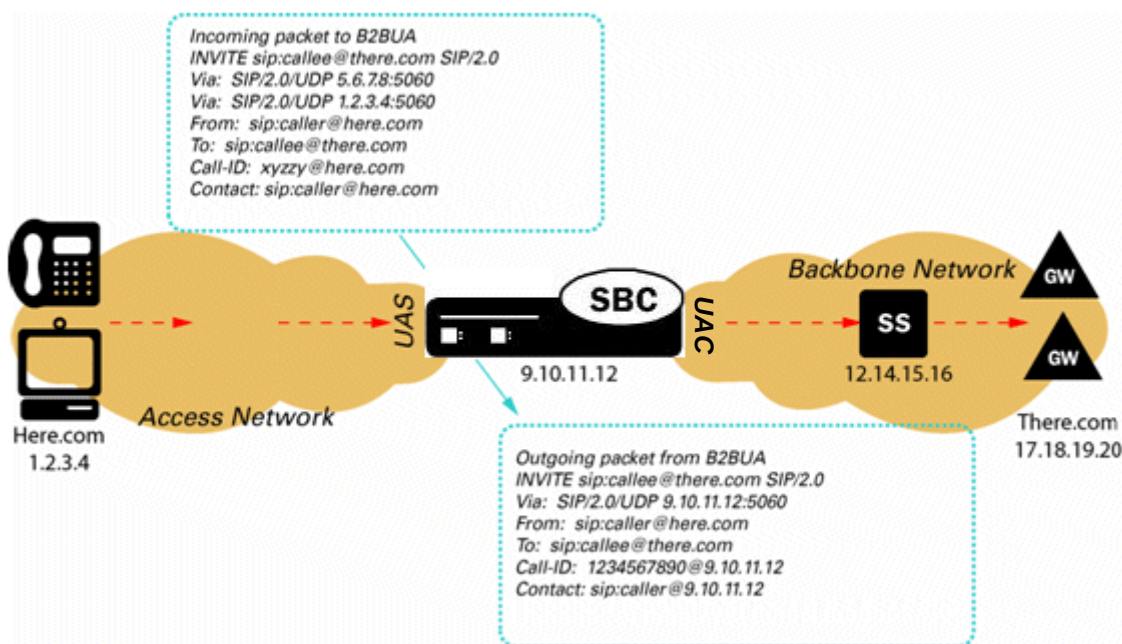
The Net-Net SBC operates as a SIP B2BUA. It terminates SIP sessions and re-originates them as new sessions as they are routed through the Net-Net SBC. For each session, it establishes NAPT translations and re-writes SDP to allow all session related media to be routed through the Net-Net SBC. It generates new call IDs and modifies SIP headers to prevent any protected SIP addresses and route information from being transmitted to external peers. The Net-Net SBC supports multiple SIP interfaces that are associated with a set of media ports, thus appearing as multiple virtual SIP gateways.



B2BUA Hosted IP Services

The Net-Net SBC acts as an outbound proxy for SIP endpoints and performs the operations required to allow UAs behind NATs to initiate and terminate SIP sessions (Hosted NAT Traversal).

The Net-Net SBC caches registration requests from SIP endpoints and forwards them to the appropriate softswitch or registrar in its backbone network. All subsequent signaling between the endpoint and the backbone network is through the Net-Net SBC. Also, all calling features such as caller ID, call waiting, three-way calling, and call transfer are all supported transparently through the Net-Net SBC.



SIP B2BUA and L3/L5 NAT

For each SIP session, the Net-Net SBC establishes NAT translations and re-writes SDP to route all session related media through the Net-Net SBC. These actions make the Net-Net SBC look like a SIP gateway. Also, the Net-Net SBC support of multiple SIP interfaces associated with different network interfaces makes it appear as multiple virtual SIP gateways.

This functionality enables the Net-Net SBC to deliver VoIP services to multiple end users, across a VPN backbone.

About SIP Interfaces

The SIP interface defines the transport addresses (IP address and port) upon which the Net-Net SBC receives and sends SIP messages. You can define a SIP interface for each network or realm to which the Net-Net SBC is connected. SIP interfaces support both UDP and TCP transport, as well as multiple SIP ports (transport addresses). The SIP interface's SIP NAT function lets Hosted NAT Traversal (HNT) be used in any realm.

SIP INVITE Message Processing

When the session agent element on the softswitch side of the message flow (ingress session agent) has the gateway contact parameter configured as an option, the Net-Net SBC looks for the URI parameter (as defined by the gateway contact parameter) in the Request-URI and decodes the gateway address.

Examples

The following example shows a SIP INVITE message from a softswitch to a Net-Net SBC.

```
INVITE sip:05030205555@ss-side-ext-address;gateway=encoded-gw-address
From: "Anonymous"<sip:anonymous@anonymous.invalid>;tag=xxxx
To: <sip:05030205555@ss-side-ext-address;user=phone>
```

The following example shows a SIP INVITE message from a Net-Net SBC to a gateway.

```
INVITE sip:05030205555@gw-ip-address SIP/2.0
From: "Anonymous"<sip:anonymous@anonymous.invalid>;tag=SDxxxx-xxxx
To: <sip:05030205555@ hostpart;user=phone>
```

The Net-Net SBC converts the `hostpart` in the To header except in the following scenarios:

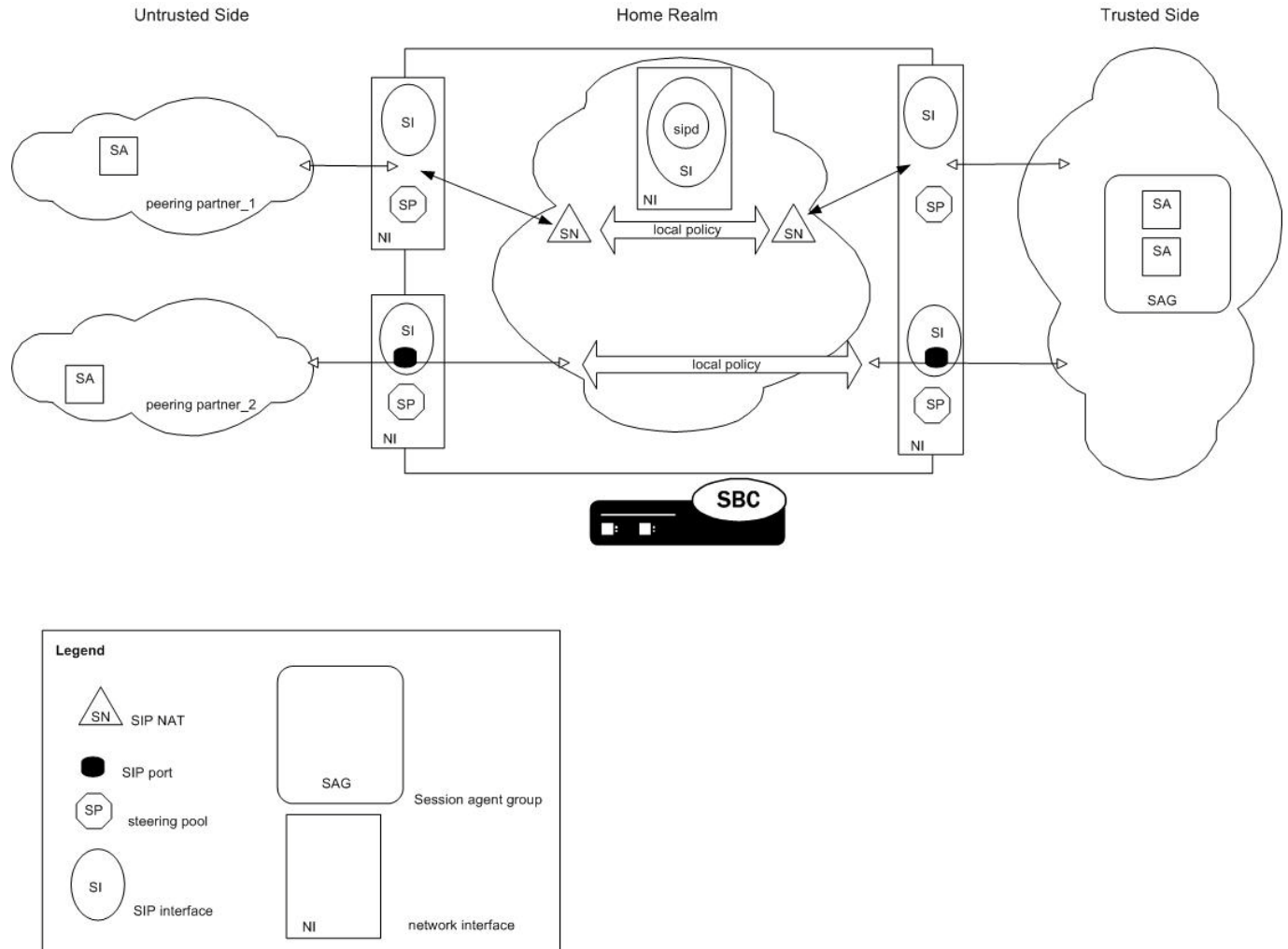
- when the original hostpart value received is an Fully Qualified Domain Name (FQDN)
- when the Net-Net SBC is configured *not* to NAT the To headers

Acme Packet recommends configuring the Net-Net SBC to NAT the To headers to ensure the security of protected addresses. Otherwise, the outgoing hostpart is set to the SIP NAT's external proxy address for the SIP NAT's external realm.

Configuring the Net-Net SBC for SIP Signaling

This section contains a diagram of a B2BUA peering environment that illustrates the Net-Net SBC components you need to configure.

Refer to the following sections for details about configuring the Net-Net SBC for SIP signaling.



Using the Net-Net Central to Configure the Net-Net SBC

You must create all new configurations and make edits to existing configurations in the Inactive configuration branch of the Net-Net Central navigation tree. You then save and activate the Inactive configuration to apply the new configurations and edits on your Net-Net SBC.

Refer to the *Net-Net Central User Guide* for detailed instructions.

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by	admin@console
Last modified date	2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2.UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Home Realm

This section explains how to configure a home realm. The home realm applies only to a SIP configuration. It represents the internal default realm or network for the Net-Net SBC and is where the Net-Net SBC's SIP proxy is located.

Overview

You primarily use a home realm when using the SIP NAT function to connect multiple realms/networks to the Net-Net SBC. You define the home realm defined as either public or private for the purposes of using the SIP NAT function. If the home realm is public, all external realms are considered private. If the home realm is private, all external networks are considered public. Usually the home realm is public.

Messages are encoded (for example, the topology is hidden) when they pass from a private to a public realm. Messages are decoded when they pass from a public realm to a private realm.

These external realms/networks might have overlapping address spaces. Because SIP messages contain IP addresses, but no layer 2 identification (such as a VLAN tag), the SIP proxy must use a single global address space to prevent confusing duplicate IP addresses in SIP URIs from different realms.

SIP NAT Function

The SIP NAT function converts external addresses in SIP URIs to an internal home realm address. Usually the external address is encoded into a cookie that is added to

the userinfo portion of the URI and the external address is replaced with a home realm address unique to the SIP NAT (the SIP NAT home address).

URIs are encoded when they pass from a private realm to a public realm. When an encoded URI passes back to the realm where it originated, it is decoded (the original userinfo and host address are restored). The encoding/decoding process prevents the confusion of duplicate addresses from overlapping private addresses. It can also be used to hide the private address when a SIP message is traversing a public network. Hiding the address occurs when it is a private address; or when the owner of the private network does not want the IP addresses of their equipment exposed on a public network or on other private networks to which the Net-Net SBC connects.

Home Realm's Purpose

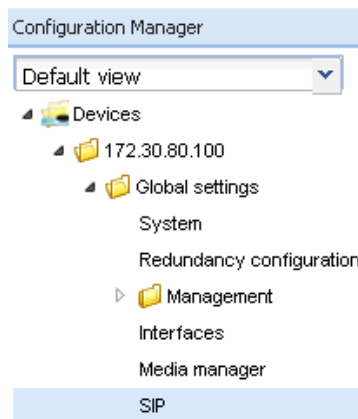
A home realm is required because the home address for SIP NATs is used to create a unique encoding of SIP NAT cookies. You can define the home realm as a network internal to the Net-Net SBC, which eliminates the need for an actual home network connected to the Net-Net SBC. You can define this virtual home network if the supply of IP addresses is limited (because each SIP NAT requires a unique home address), or if all networks to which the Net-Net SBC is connected must be private to hide addresses.

For example, you can define a public home realm using the loopback network (127.0.0.0) and using the home realm address prefix (for example, 127.0.0.0/8) for encoding addresses that do not match (all addresses outside 127.0.0.0/8) in SIP NAT cookies. The SIP NAT address prefix field can be used to accomplish this while keeping the ability to define an address prefix for the realm for ingress realm determination and admission control. By defining the SIP NAT address prefix as 0.0.0.0, the home realm address prefix is used to encode addresses that do not match.

Configuring the Home Realm

This section describes how to configure the SIP home realm on your Net-Net SBC from the Net-Net Central.

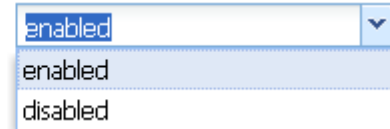
1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.

4. **SIP enabled**—Select enabled from the combination box to enable the SIP functionality on this Net-Net SBC.

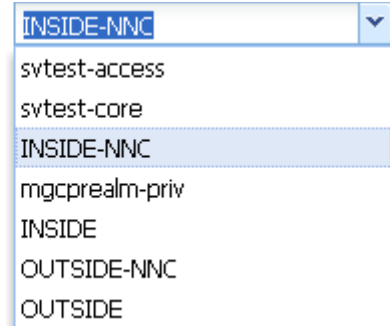
SIP enabled



A screenshot of a web-based configuration interface. It shows a label 'SIP enabled' followed by a dropdown menu. The dropdown menu is open, showing two options: 'enabled' (which is highlighted with a blue background) and 'disabled'.

5. **Realm name**—Select the SIP home realm from the list of preconfigured realms.

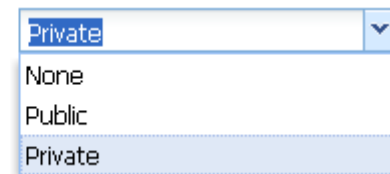
Realm name



A screenshot of a web-based configuration interface. It shows a label 'Realm name' followed by a dropdown menu. The dropdown menu is open, showing a list of realm names: 'svtest-access', 'svtest-core', 'INSIDE-NNC' (highlighted with a blue background), 'mgcprealm-priv', 'INSIDE', 'OUTSIDE-NNC', and 'OUTSIDE'.

6. **Realm type**—Select the realm type, which indicates the SIP NAT mode.
- **Public**—Indicates the subnet defined in the addr-prefix-id field of the home realm is public and the subnet defined in the addr-prefix-id field of all external realms identified in the SIP NAT are private networks. IP addresses are encoded in SIP messages received from the external realm defined by the SIP NAT. The IP addresses are decoded in messages that are sent to the realm.
 - **Private**—Indicates the subnet defined in the addr-prefix-id field of the home realm is private and the subnet defined in the addr-prefix-id field of all external realms identified in the SIP NAT are public networks. IP addresses are encoded in SIP messages sent to the external realm defined by the SIP NAT and decoded in messages received from the realm.
 - **None**—No SIP NAT function is necessary.

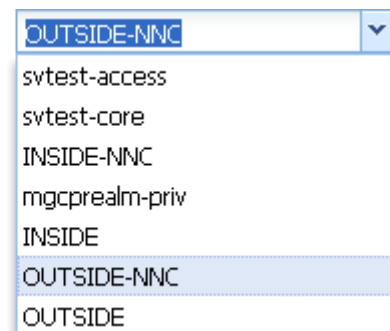
Realm type



A screenshot of a web-based configuration interface. It shows a label 'Realm type' followed by a dropdown menu. The dropdown menu is open, showing three options: 'Private' (highlighted with a blue background), 'None', and 'Public'.

7. **Default egress realm**—Select the default egress realm from the combination box of preconfigured realms; *blank* is a valid selection.

Default egress realm



A screenshot of a web-based configuration interface. It shows a label 'Default egress realm' followed by a dropdown menu. The dropdown menu is open, showing a list of realm names: 'svtest-access', 'svtest-core', 'INSIDE-NNC', 'mgcprealm-priv', 'INSIDE', 'OUTSIDE-NNC' (highlighted with a blue background), and 'OUTSIDE'.

8. Click **Apply**.

SIP Interface

This section explains how to configure a SIP interface. The SIP interface defines the transport addresses (IP address and port) upon which the Net-Net SBC receives and sends SIP messages.

Overview

The SIP interface defines the signaling interface. You can define a SIP interface for each network or realm to which the Net-Net SBC is connected. SIP interfaces support both UDP and TCP transport, as well as multiple SIP ports (transport addresses). The SIP interface also lets Hosted NAT Traversal (HNT) be used in any realm.

The SIP interface configuration process involves configuring the following features:

- address and transport protocols (SIP ports)
- redirect action
- proxy mode
- trust mode

About SIP Ports

A SIP port defines the transport address and protocol the Net-Net SBC will use for a SIP interface for the realm. A SIP interface will have one or more SIP ports to define the IP address and port upon which the Net-Net SBC will send and receive messages. For TCP, it defines the address and port upon which the Net-Net SBC will listen for inbound TCP connections for a specific realm.

You need to define at least one SIP port, on which the SIP proxy will listen for connections. If using both UDP and TCP, you must configure more than one port. For example, if a call is sent to the Net-Net SBC using TCP, which it needs to send out as UDP, two SIP ports are needed.

Preferred SIP Port

When a SIP interface contains multiple SIP ports of the same transport protocol, a preferred SIP port for each transport protocol is selected for outgoing requests when the specific SIP port cannot be determined. When forwarding a request that matched a cached registration entry (HNT or normal registration caching), the SIP port upon which the original REGISTER message arrived is used. Otherwise, the preferred SIP port for the selected transport protocol is used. When selecting the preferred SIP port, the default SIP port of 5060 will be selected over other non-default ports.

For SIP interfaces using the SIP NAT function, the preferred SIP port address and port will take precedence over the external address of the SIP NAT when they do not match. If both TCP and UDP SIP ports are defined, the address and port of the preferred UDP port is used.

Proxy Mode

The Net-Net SBC's proxy mode determines whether it forwards requests received on the SIP interface to target(s) selected from local policy; or sends a redirect response to the previous hop. Sending the redirect response causes the previous hop to contact the targets directly.

If the source of the request matches a session agent with a proxy mode already defined, that mode overrides the proxy mode defined in the SIP interface.

You can configure the proxy mode to use the Record-Route option. Requests for stateless and transaction operation modes are forwarded with a Record-Route header that has the Net-Net SBC's addresses added. As a result, all subsequent requests are routed through the Net-Net SBC.

Redirect Action

The redirect action is the action the SIP proxy takes when it receives a SIP Redirect (3xx) response on the SIP interface. If the target of the request is a session agent with redirect action defined, its redirect action overrides the SIP interface's.

You can set the Net-Net SBC to perform a global redirect action in response to Redirect messages. Or you can retain the default behavior where the Net-Net SBC sends SIP Redirect responses back to the previous hop (proxy back to the UAC) when the UAS is not a session agent.

The default behavior of the Net-Net SBC is to recurse on SIP Redirect responses received from the user agent server (UAS) and send a new request to the Contact headers contained in the SIP Redirect response.

Instead of this default behavior, the Net-Net SBC can proxy the SIP Redirect response back to the user agent client (UAC) using the value in the session agent's redirect action field (when the UAS is a session agent). If there are too many UASes to define as individual session agents or if the UASs are HNT endpoints, and SIP Redirect responses need to be proxied for UASs that are not session agents; you can set the default behavior at the SIP Interface level.

SIP MADDR Resolution

Addresses found in SIP contact headers, or in the **maddr** (multicast address) parameter of SIP 3xx REDIRECT messages can be resolved as a host address, or a session agent name, or a session agent group (SAG) names.

Support for SAG-based resolution is provided by a new **sip-config** parameter, **Lookup SAG name on redirect**. By default, SAG lookup is disabled, providing compatibility with prior releases.

The following sample SIP REDIRECT and ACLI configuration fragment illustrate enhanced processing.

Status-Line: SIP/2.0 302 Moved

Message Header

```
Via: SIP/2.0/UDP
192.168.200.224:5060;branch=z9hG4bKa0fs40009o90sc8oo780.1
From: <sip:1111@192.168.1.222:6000>;tag=1
To: sut <sip:2223@192.168.1.224:5060>;tag=11
Call-ID: 1-28515@192.168.1.222
CSeq: 1 INVITE
Contact: <sip:1111@192.168.1.223;maddr=test.acmepacket.com>
Privacy: user,id,critical;session
P-Preferred-Identity: sipp <sip:sipp@192.168.200.222:5060>
P-Asserted-Identity: abc.com
Subject: abc
Proxy-Require: privacy,prack,abc
Content-Length: 0
```

Session agent group

```
Group name: test.acmepacket.com
```

Description
 State: enabled
 Protocol: SIP
 Session agent selection strategy: Hunt

...

In this case, when the SBC receives the 302, it resolves the information from **maddr** to a SAG name. In the above example, it will resolve to the configured SAG – test.acmepacket.com. The destinations configured in SAG test.acmepacket.com will be used to route the call.

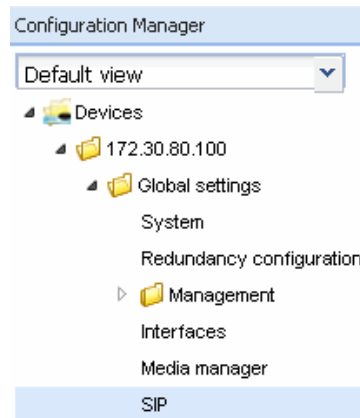
SAG-based address resolution is based on the following set of processing rules:

- When the Contact URI does not have an maddr parameter, and the hostname is not an IP Address, the Net Net SBC will look for a SAG matching the hostname. 192.168.200.223
- When the Contact URI has an maddr parameter that contains an IP address, the Net Net SBC will not look for a SAG; it will use the IP address as the target/next-hop.
- When the Contact URI has an maddr parameter that contains a non-IP-address value, the Net Net SBC will look for a SAG matching the maddr parameter value.

Configuring SAG-Based MADDR Resolution

To configure the Net-Net SBC to perform SAG-based maddr resolution:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **Lookup SAG name on redirect**—Select enabled SAG-based MADDR resolution.

Lookup SAG name on redirect

5. Click **Apply**.

Trust Mode

The Net-Net SBC supports the Calling Identity privacy requirements based on RFC 3323 and RFC 3325. The trust mode in the SIP interface determines whether the source and destination of a request is a trusted entity. With the implementation of this feature, the Net-Net SBC can understand and support the privacy headers and provide the capability for anonymous packets.

The Net-Net SBC, which acts as a boundary device between the trusted platform and the untrusted Internet, understands the following headers:

- Privacy Header
- P-Asserted-Identity Header
- P-Preferred-Identity Header

Depending on the value of these headers and the mode in which the Net-Net SBC is being operated (B2BUA or the proxy), the appropriate actions are performed.

About the Process

On receiving a message, the Net-Net SBC checks whether the message source is trusted or not. It checks the SIP interface's trust mode value and, if the source is a session agent, the session agent's trust me value. Depending on these values, the Net-Net SBC decides whether the request's or response's source is trusted. If it receives message from a trusted source and the message contains the P-Asserted-Identity header field, the Net-Net SBC passes this message to the outgoing side. The outgoing side then decides what needs to be done with this request or response.

If the request or the response is received from an untrusted source, the Privacy header value is *id* (privacy is requested), and the P-Asserted-Identity header field is included, the Net-Net SBC strips the Privacy and the P-Asserted-Identity headers and passes the request or the response to the outgoing side.

If the request or the response contains the P-Preferred-Identity header and the message source is untrusted, the Net-Net SBC strips the P-Preferred-Identity header from the request or the response and passes the message to the outgoing side.

If the source is trusted or privacy is not requested (the value of the Privacy Header is not *id*) and the request or the response contains the P-Preferred-Identity header, the Net-Net SBC performs the following actions:

- inserts the P-Asserted-Identity header field with the value taken from the P-Preferred-Identity header field
- deletes the P-Preferred-Identity header value
- passes this request or the response to the Outgoing side for the appropriate action, depending on the whether the destination is trusted or not

After the Net-Net SBC passes the request or the response to the outgoing side, it checks whether the destination is trusted by checking the SIP interface's trust mode value and the session agent's trust me value (if the destination is configured as session agent).

- The destination is trusted

The Net-Net SBC does nothing with the request or the response and passes it to the destination. If the P_Asserted_Identity headers are present, they are passed to the session agent (if the destination is configured as session agent).

- The destination is untrusted

The Net-Net SBC looks at the value of the Privacy header. If set to *id*, the Net-Net SBC removes all the P-Asserted-Identity headers (if present). It strips the Proxy-Require header if it is set to *privacy*. The Net-Net SBC also sets the From field of SIP header to *Anonymous* and strips the Privacy header.

If the Privacy header is set to *none*, the Net-Net SBC does not remove the P-Asserted-Identity header fields.

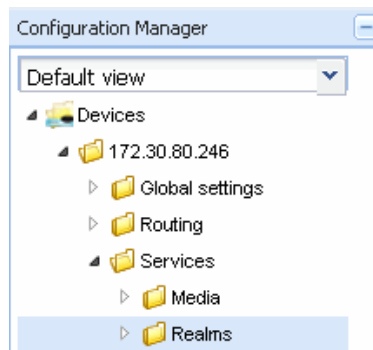
If there is no Privacy header field, the SD will not remove the P-Asserted-Identity headers.

To implement this feature, you need to configure the session agent's trust me parameter to enabled (if the message source is a session agent) and the SIP interface's trust mode to the appropriate value.

Configuring a SIP interface

This section describes how to configure a SIP interface.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the table and click **Edit**.
4. Click the Services and Protocols tab. The Services and Protocols table appears.
5. Click **Add**.



Services and Protocols	
Service type	Identifier
sipInterface	ID18



6. Select sipInterface from the combination box.

The screenshot shows a configuration window titled "realmConfig". Under the "Service" label, a dropdown menu is open, displaying three options: "sipInterface" (which is highlighted), "H323StackConfig", and "mgcpConfig".

7. **Description**—Enter an optional descriptive string to identify the configuration and its function.

Description

An empty rectangular text input field for entering a description.

8. **State**—Set this SIP interface to enabled.

State

A dropdown menu for the "State" field. The "enabled" option is selected and highlighted in blue. Other visible options include "disabled".

9. **Trust mode**—Select which entities are considered trusted from the combination box.

Trust mode

A dropdown menu for the "Trust mode" field. The "all" option is selected and highlighted in blue. Other visible options include "registered", "agents-only", and "realm-prefix".

- **none**—Trust nothing
- **agents-only**—Trust only trusted session agents
- **registered**—Trust only trusted session agents or registered endpoints
- **realm-prefix**—Trust only trusted session agents or address matching realm prefix
- **all**—Trust all previous and next hops except untrusted session agents

10. **Proxy**—Set the proxy mode action of this SIP interface.

Modes

Proxy

Redirect

Contact header routing mode

A dropdown menu for the "Proxy" field. The "Redirect" option is selected and highlighted in blue. Other visible options include "Proxy" and "Record-Route".

- **proxy**—Forward all SIP requests to be selected again

- **redirect**—Send a SIP 3xx redirect response with the selected target(s) in the Contact header
- **record-route**—Forward requests to selected target(s) and insert a Record-Route header with the Net-Net SBC's address.

11. **Redirect**—Set the redirect action of this SIP interface.

Modes

Proxy	
Redirect	
Contact header routing mode	Proxy Recurse

- **proxy**—Send the SIP request back to the previous hop
- **recurse**—Recurse on the Contacts in the response

The designated proxy action will apply to SIP 3xx responses received from non-session agents and to 3xx responses received from session agents without configured SIP Redirect message actions (for example, session agents without values for the redirect action field).

12. **Contact header routing mode**—Determines how the contact address from a private network is formatted.

Modes

Proxy	Redirect
Redirect	Recurse
Contact header routing mode	none strict-route none maddr loose-route

For example, whether a maddr parameter equal to the Net-Net system's SIP proxy needs to be added to a URI present in a Contact header.

- **none**—The address portion of the header becomes the public address of that private realm
- **maddr**—The address portion of the header will be set to the IP address of the Net-Net system's B2BUA
- **strict**—The contents of the Request-URI is destroyed when a Record-Route header is present.
- **loose-route**—The Record-Route header is included in a Request, which means the destination of the request is separated from the set of proxies that need to be visited along the way.

13. **Scheme**—Select the type of SIP URI scheme to use from the combination box.

Set to enabled to convert SIP URIs to *tel* (resources identified by telephone numbers) URIs.

If enabled, the requests generated on this SIP interface by the Net-Net SBC will have a *tel* URI scheme instead of the SIP URI scheme. Only the Request, From, and To URIs are changed to the *tel* scheme. After the dialog is established, the URIs are not changed.

14. **Fully qualified domain name**—Enter an FQDN domain to replace the host portion of a SIP URI.

URI - To, from, and request headers

Scheme	enabled
Fully qualified domain name	eng1.acmepacket.com

The requests generated by the Net-Net SBC on this SIP interface will have the host part of the URI set to this FQDN value. Only the Request, To, and From URIs are changed. After the dialog is established, the URIs are not changed.

15. At the top of the content area, click the Registration tab.
16. **Endpoint type supported by NAT traversal**—Set the type of HNT used from the combination box.

NAT traversal

Endpoints type supported by NAT traversal	none
Endpoint to SD registration interval UDP (sec)	none
Endpoint to SD registration interval TCP (sec)	90

- **none**—HNT function disabled for SIP (default value).
 - **rport**—SIP HNT function only applies to endpoints that include the rport parameter in the Via header. HNT applies when the sent-by of the topmost VIA matches the Contact-URI host address, both of which must be different from the received Layer 3 address.
 - **always**—SIP HNT applies to requests when the sent-by of the topmost VIA matches the Contact-URI host address, both of which must be different from the received Layer 3 address. (Even when the rport parameter is not present.)
17. **Endpoint to SD registration interval UDP (sec)**—Set the expiration time in seconds for an HNT endpoint's cached registration entry.
 - minimum: zero (0) seconds
 - maximum: 999999999 seconds

Acme Packet recommends setting the NAT interval to one-third of the NAT binding lifetime. A NAT binding lifetime is the network connection inactivity timeout. The value is configured (or hardwired) in the NAT device (firewall). This timer is used to cause the UA to send REGISTER messages frequently enough to retain the port binding in the NAT. Retaining the binding lets inbound requests to be sent through the NAT.

18. **SD to registrar registration interval (sec)**—Set the minimum expiration time in seconds for an HNT endpoint's cached registration entry.

The value you enter here sets the minimum registration expiration time in seconds for HNT registration caching.

- minimum: zero (0) seconds
- maximum: 999999999 seconds

The default value is 300.

This value defines the minimum expiration value the Net-Net SBC places in each REGISTER message it sends to the real registrar. In HNT, the Net-Net SBC caches the registration after receiving a response from the real registrar and sets the expiration time to the NAT interval value.

Some UAs might change the registration expiration value they use in subsequent requests to the value specified in this field. This change causes the Net-Net SBC to send frequent registrations on to the real registrar.

19. Click the Registration tab at the top of the content area.
20. **Send transaction to registrar**—Set whether routing to the registrar is enabled.

Registration handling

Send transactions to registrar	enabled	▼
Registration caching	enabled	▼
SD to registrar registration interval (sec)	3600	

Enable routing to the registrar to send all requests that match a cached registration to the destination defined for the registrar host; used when the Request-URI matches the registrar host value or the registrar domain value, not the Net-Net SBC's address. Because the registrar host is the real registrar, it should send the requests back to the Net-Net SBC with the Net-Net SBC's address in the Request-URI. By default, this field is set to *disabled*.

For example, you should enable routing to the registrar if your network uses a Net-Net SBC and needs requests to go through its service proxy, which is defined in the registrar host field.

21. **Registration caching**—Select which endpoints' registrations are cached from the combination box.

Enable for use with all UAs, not just those that are behind NATs. By default, this field is set to *disabled*. If enabled, the Net-Net SBC caches the Contact header in the UA's REGISTER request when it is addressed to one of the following:

- Net-Net SBC
- registrar domain value
- registrar host value

The Net-Net SBC then generates a Contact header with the Net-Net SBC's address as the host part of the URI and sends the REGISTER to the destination defined by the registrar host value.

22. **SD to registrar registration interval (sec)**—Enter Net-Net SBC's cached registration entry interval for a non-HNT endpoint.

Enter the expiration time in seconds that you want the Net-Net SBC to use in the REGISTER response message sent back to the UA. The UA then refreshes its registration by sending another REGISTER message before that time expires.

- minimum: zero (0) seconds

A registration interval of zero causes the Net-Net SBC to pass back the expiration time set by and returned in the registration response from the registrar.

- maximum: 999999999 seconds

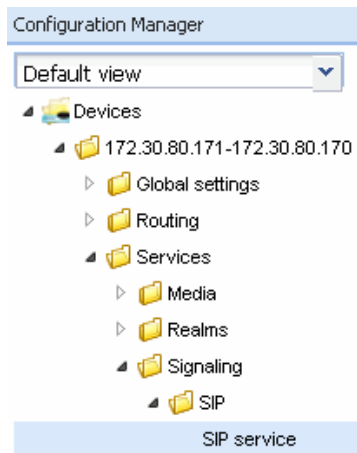
The default value is 3600 seconds (one hour). If the expiration time you set is less than the expiration time set by and returned from the real registrar, the Net-Net SBC responds to the refresh request directly rather than forwarding it to the registrar.

Although the registration interval applies to non-HNT registration cache entries, and the loosely related NAT interval applies to HNT registration cache entries, you can use the two in combination. Using a combination of the two means you can implement HNT and non-HNT architectures on the same Net-Net SBC. You can then define a longer interval time in the registration interval field to reduce the network traffic and load caused by excess REGISTER messages because there is no NAT binding to maintain.

Configuring SIP Ports

To configure SIP ports:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Select a SIP Interface and click **Edit**.
4. Scroll to SIP signaling interface table and click **Add**.
The Add SIP signaling interface window appears.
5. **IP addresss**—Enter the IP address of this SIP signaling interface.
6. **IP port**—Enter the port number of this SIP signaling interface.
 - minimum: 1025
 - maximum: 65535

7. **Transport protocol**—Enter the transport protocol this SIP signaling interface uses.

Add SIP signaling interface

*IP address: 0.0.0.0 (Unique required. Enter a valid IP address)

*IP port: 5060 (Unique required. Range:1025..65535,..0..0. Default:5060)

*Transport protocol: SCTP (Unique required. Default:UDP)

UDP
TLS
SCTP
TCP

- **TCP**—Provides a reliable stream delivery and virtual connection service to applications through the use of sequenced acknowledgment with the retransmission of packets when necessary.
- **UDP**—Provides a simple message service for transaction-oriented services. Each UDP header carries both a source port identifier and destination port identifier, allowing high-level protocols to target specific applications
- **TLS**—Protects user and network privacy by providing authentication and guaranteeing the integrity for communications between the Net-Net SBC and other devices, including other Net-Net SBCs
- **STCP**—SCTP is used as the transport method. SCTP supports multi-homing and multi-streaming.

Multi-homing refers to instances when multiple IP addresses are assigned to a host on the network. Typically, this arrangement entails a host that has multiple network interface cards.

Multi-streaming refers to the ability to partition data within an association into multiple logical communication channels. Each logical channel—or stream—has the property of an independent sequenced delivery. This means that data loss on one stream has no impact on delivery on other streams.

If you choose TLS, the TLS profile textbox becomes activated.

8. **TLS profile**—Choose the profile name from the TLS profile combination box.

TLS profile

INSIDE

INSIDE

9. Click **Apply** to continue.

Timers

This section explains how to configure SIP retransmission and expiration timers.

Overview

SIP timers define the transaction expiration timers, retransmission intervals when UDP is used as a transport, and the lifetime of dynamic TCP connections. The retransmission and expiration timers correspond to the timers defined in RFC 3261.

- **init timer**—Is the initial request retransmission interval. It corresponds to Timer T1 in RFC 3261

This timer is used when sending requests over UDP. If the response is not received within this interval, the request is retransmitted. The retransmission interval is doubled after each retransmission.

- **max timer**—Is the maximum retransmission interval for non-INVITE requests. It corresponds to Timer T2 in RFC 3261

The retransmission interval is doubled after each retransmission. If the resulting retransmission interval exceeds the max timer, it is set to the max timer value.

- **trans expire**—Is the transaction expiration timer. This value is used for timers B, D, F, H and J as defined in RFC 3261
- **invite expire**—Defines the transaction expiration time for an INVITE transaction after a provisional response has been received. This corresponds to timer C in RFC 3261

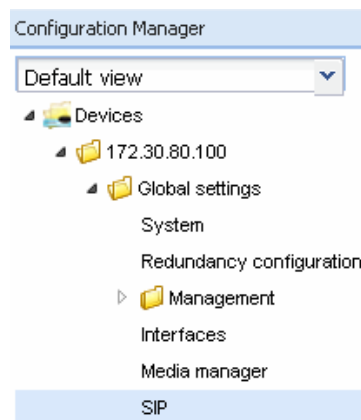
If a final response is not received within this time, the INVITE is cancelled. In accordance with RFC 3261, the timer is reset to the invite expire value when any additional provisional responses are received.

- **inactive dynamic conn timer**—Defines the idle time (no active sessions and no SIP messages sent or received) of a dynamic TCP connection before the connection is torn down. There is no timer in RFC 3261 corresponding to this function.

Configuring SIP Timers

This section describes how to configure SIP Timers on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to Timers in the content area.
5. **Response to INVITE initial timeout - RFC 3261 timer T1 (msec)**—Enter the initial timeout (T1) value for SIP in milliseconds. This value is applied to responses to INVITE requests and any UDP SIP requests. The available values include:
 - minimum: 0 milliseconds
 - maximum: 999999999 milliseconds
6. **Max. retransmit interval: INVITE responses and non-INVITE requests - RFC 3261 timer T2 (msec)**—Enter the maximum transmission timeout (T2) for SIP in milliseconds.

When sending SIP over UDP, a re-transmission timer is used. If the timer expires and the message is re-transmitted, the re-transmission timer is then set to twice the previous value (but will not exceed the maximum timer value). Using the default values of 500 milliseconds and SBC milliseconds, the re-transmission timer is 0.5, then 1, 2, and finally 4. The incrementing continues until the transmission expire timer activates.

- minimum: 0 milliseconds
 - maximum: 999999999 milliseconds
7. **Completion of SIP transactions timeout - RFC 3261 timers B, D, F, H, J (sec)**—Enter the transaction expire timeout value (Timer B) in seconds to set the time for SIP transactions to live. The same value is used for Timers D, F, H and J.
 - minimum: 0 seconds
 - maximum: 999999999 seconds
 8. **Completion of SIP client transaction timeout - RFC 3261 timer C (sec)**—Enter the invite expire timeout value (Timer C) in seconds to indicate the time for SIP client transaction will live after receiving a provisional response.
 - minimum: 0 seconds
 - maximum: 999999999 seconds
 9. **Inactive TCP/TLS-based session timeout (sec)**—Enter the inactive dynamic connection value in seconds to set the time limit for inactive dynamic connections.

If the connection between the SIP proxy and a session agent is dynamic (for example, through dTCP), and the connection has been idle for the amount of time specified here, the SIP proxy breaks the connection.

- minimum: 1 second
- maximum: 999999999 seconds

Timers

Response to INVITE initial timeout - RFC 3261 timer T1 (msec)	500
Max. retransmit interval: INVITE responses and non-INVITE requests - RFC 3261 timer T2 (msec)	4000
Completion of SIP transactions timeout - RFC 3261 timers B,D,F,H,J (sec)	32
Completion of SIP client transaction timeout - RFC 3261 timer C (sec)	180
Inactive TCP/TLS-based session timeout (sec)	32

10. Click **Apply** to complete configuring SIP timers.

SIP Options Tag Handling

This section explains how to configure SIP options on a global or per-realm level and how to specify whether the feature treatment applies to traffic inbound to or outbound from a realm, or both.

SIP extensions that require specific behavior by UAs or proxies are identified by option tags. Option tags are unique identifiers used to designate new options (for example, extensions) in SIP. These option tags appear in the Require, Proxy-Require, and Supported headers of SIP messages.

Option tags are compatibility mechanisms for extensions and are used in header fields such as Require, Supported, Proxy-Require, and Unsupported in support of SIP.

The option tag itself is a string that is associated with a particular SIP option (i.e., an extension). It identifies this option to SIP endpoints.

Overview

The SIP specification (RFC 3261) requires that the Net-Net SBC B2BUA reject any request that contains a Require header with an option tag the Net-Net system does not support. However, many of these extensions operate transparently through the Net-Net system's B2BUA. You can configure how SIP defines the Net-Net system's B2BUA treatment of specific option tags.

Also, there might be certain extensions that an endpoint indicates support for by including the option tag in a Supported header. If you do not want a given extension used in your network, the you can configure SIP option tag handling to remove the undesired option tag from the Supported header. You can also specify how option tags in Proxy-Require headers are to be treated.

Configuration Overview

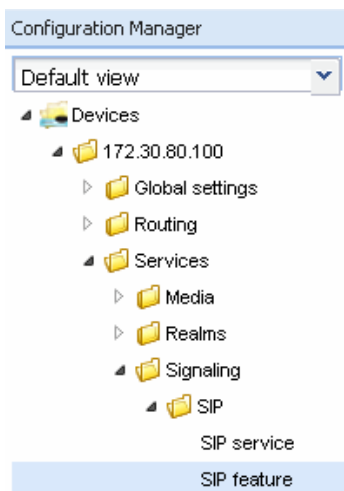
You configure the SIP feature element to define option tag names and their treatment by the Net-Net system when the option tag appears in a Supported header, a Require header, and a Proxy-Require header. If an option tag is encountered that is not configured as a SIP feature, the default treatments apply. You only need to configure option tag handling in the SIP feature element when non-default treatment is required.

You can specify whether a SIP feature should be applied to a specific realm or globally across realms. You can also specify the treatment for an option based on whether it appears in an inbound or outbound packet. Inbound packets are those that are coming from a realm to the Net-Net SBC and outbound packets are those which are going from the Net-Net SBC to the realm.

Configuring SIP Options Tag Handling

This section describes how to configure SIP option tags on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Servicess > Signaling > SIP.



2. Click SIP feature. The SIP feature table appears in the content area.
3. Under the SIP option tag table, click **Add**.
4. **Realm**—Select the realm from the combination box to which these tags apply
5. **Tag name**—Enter the SIP option tag name.

The 'Add SIP Feature' dialog box has two input fields:

- *Realm: A dropdown menu showing 'INSIDE-NNC' with a '(Unique required)' note.
- *Tag name: A text input field containing 'newtag' with a '(Unique required)' note.

 At the bottom are 'Apply' and 'Cancel' buttons.

6. Click **Apply**.
7. Select the Tag name from the table and click **Edit**.
8. **Support header**—Select the support header action for inbound tags from the combination box.
 - Pass—B2BUA should include the tag in the corresponding incoming message.
 - Strip—Tag should be excluded in the incoming message. Use strip mode to not use the extension.
9. **Require header**—Select the require header action for inbound tags from the combination box.
 - Pass—B2BUA should include the tag in the corresponding incoming message.
 - Reject—B2BUA should reject the request with a 420 (Bad Extension) response. The option tag will be included in an Unsupported header in the reject response.
10. **Proxy-require header**—Select the proxy-require header action for inbound tags from the combination box.

- Pass—B2BUA should include the tag in the corresponding incoming message.
 - Reject—B2BUA should reject the request with a 420 (Bad Extension) response. The option tag will be included in an Unsupported header in the reject response.
11. **Support header**—Select the support header action for outbound tags from the combination box.
- Pass—B2BUA should include the tag in the corresponding outgoing message.
 - Strip—Tag should be excluded in the outgoing message. Use strip mode to not use the extension.
12. **Require header**—Select the require header action for outbound tags from the combination box.
- Pass—B2BUA should include the tag in the corresponding outgoing message.
 - Reject—B2BUA should reject the request with a 420 (Bad Extension) response. The option tag will be included in an Unsupported header in the reject response.
13. **Proxy-require header**—Select the proxy-require header action for outbound tags from the combination box.
- Pass—B2BUA should include the tag in the corresponding incoming message.
 - Reject—B2BUA should reject the request with a 420 (Bad Extension) response. The option tag will be included in an Unsupported header in the reject response.

*Tag name	newtag
*Realm	INSIDE-NNC ▼

Inbound tags

Support header	Pass ▼
Require header	Reject ▼
Proxy-require header	Pass ▼

Outbound tags

Support header	Pass ▼
Require header	Reject ▼
Proxy-require header	Pass ▼ Pass Reject

14. Click **Apply**.

SIP Options

This section explains how you can configure a limited list of specialized SIP features and/or parameters called options. The options described here were developed to meet specific needs not addressed by the standard SIP configuration parameters. Not all users have a need for these options.

Note: Acme Packet recommends checking with your Acme Packet representative before applying any of these options.

Overview

You can configure options for the SIP configuration and SIP interface. Both elements include a parameter (options) that you use to configure the options described in the following section.

Global SIP Options

The following table lists the SIP options supported by the Net-Net SBC.

Option	Description
add-error-to-tag=no	If present (even when set to no), suppresses the addition of an Acme tag on 3xx-6xx responses.
add-prov-to-tag=no	Prevents the Net-Net SBC from adding a tag parameter to the To header (to-tag) to non-100 provisional responses to INVITE requests. Used when a provisional (101-199) response is received from the UAS on a client transaction without a to-tag. By default, the Net-Net SBC adds the tag cookie in the response (as though it had a tag) sent back to the UAC for the associated server transaction. When you include this option in the SIP configuration, and the response from the UAS does not have a to-tag, the response forwarded to the UAC will not have a to-tag.
add-reg-expires	Causes an Expires header to always be included in a REGISTER response with the registration caching and HNT traversal functions of the Net-Net SBC. Use for endpoints that do not understand the Expires parameter in the Contact header.
add-ruri-user=<methods>	Causes a userinfo portion to be added to a Request-URI when one is not present. Used to support the OKI phone, which registers a Contact of just an IP-Address but rejects initial INVITEs if the Request_URI does not have a userinfo part. <methods> is a comma-separated list of methods to which the option should apply. If more than one method is listed, the list must be enclosed in quotes. This option only applies to out-of-dialog requests (no tag parameter in the To header). However, if ACK is listed, it will apply to all ACK requests because an ACK is always supposed to have a to-tag.
allow-notify-no-contact	Prevents the Net-Net SBC from rejecting NOTIFYs with a 400 Bad Request response. NOTIFY requests without Contact header are allowed to pass through the Net-Net SBC instead.
call-id-host=<host>	Causes the Net-Net SBC to include a host part (ID@host) in the Call-ID it generated. <host> is the hostname (or IP address) that is to appear in the host part of the Call-ID. If not specified, the SIP port address is used.

Option	Description
contact-endpoint=<param-name>	Defines a URL parameter to report the real Contact address of an endpoint in a REGISTER message forwarded to a registrar; when the Net-Net is caching registration. (plain or HNT). If <param-name> is not specified, the default value endpoint is used. This parameter is added as a URL parameter in the Contact on the REGISTER message. In order for the registration cache to work properly, the softswitch/registrar is expected to include the endpoint parameter in the Request-URI of a SIP request it forwards to the address-of-record.
contact-firewall=<param-name>	Defines a URL parameter to report the NAT between the Net-Net SBC and the real Contact address of an endpoint in a REGISTRAR message forwarded to a registrar when the Net-Net SBC is doing registration caching for NHT. If <param-name> is not specified, the default value firewall is used. This parameter will be added as a URL parameter in the Contact on the REGISTER message. In order for the registration cache to work properly, the softswitch/registrar is expected to include the endpoint parameter in the Request-URI of any SIP request it forwards for the address-of-record.
disable-privacy	Prevents the change of the P-Preferred-Identity to P-Asserted-Identity and lets the P-Preferred-Identity go through unchanged.
drain-sendonly	Causes the Net-Net SBC to examine the SDP attributes and change sendonly mode to sendrecv. This causes the endpoint receiving the SDP to send RTP, which is required for HNT traversal endpoints to work with media servers. The Net-Net SBC sets up the flow so that RTP coming from the endpoint are dropped to prevent the UA that sent the sendonly SDP from receiving packets. See the option video-sbc-session also.
encode-contact=<prefix>	Causes the Net-Net SBC to encode Contact addresses into the userinfo part of the URI. It applies only to Contact address that usually get the maddr parameter. Use when the Net-Net SBC needs requests sent to the URI in the Contact sent instead to the Net-Net SBC. The host part of the URI will have the Net-Net SBC's address. The <prefix> serves as a place between the original userinfo and the encoded address. If a <prefix> is specified, a default of +SD is used. Without this option, the Net-Net SBC adds a maddr parameter.
fix-to-header	For requests that have the Net-Net- SD's address in both the Request-URI and the To-URI, it sets the hostport of the To-URI to a local policy's next hop target on out-of-dialog requests (no to-tag). This is the default IWF behavior, even without this option configured.

Option	Description
<code>forward-reg-callid-change</code>	Addresses the case when an endpoint reboots and performs a third party registration before its old registration expires. During this re-registration, the contact header is the same as it was pre-reregistration. As a consequence of the reboot, the SIP Call-ID changes. In this situation, the Net-Net SBC does not forward the REGISTER to the registrar, because it believes the endpoint is already registered, based on a previous registration from the same Contact: header URI. To remedy this problem, the Net-Net SBC now keeps track of the Call-ID in its registration cache. A new option in the SIP interface configuration element forces the Net-Net SBC to forward a REGISTER message to the registrar when the Call-ID header changes in a REGISTER message received from a reregistering UAC.
<code>global-contact</code>	Addresses interoperability in the Dialog and Presence event packages that are used in hosted PBX and IP Centrex offerings. This option enables persistent URIs in the Contact headers inserted into outgoing SIP messages. If this option is not used, URIs placed in the Contact header of outgoing messages are only valid within the context of the dialog to which the message is associated.
<code>load-limit=<cpu percentage></code>	Defines the CPU usage percentage at which the Net-Net SBC should start rejecting calls. Default value is 95%.
<code>max-register-forward=<value></code>	Defines a limit (as assigned in the value field) of REGISTERs to be forwarded to the registrar. During each second, the sipd counts how many REGISTERs have been sent to the registrar. It checks the threshold when it receives a REGISTER from the UA and determines that less than half the real registration lifetime is left. If the number of REGISTERs forwarded (new and updates) in the current second exceeds the configured threshold, it will respond to the UA from the cache.
<code>max-register-refresh=<value></code>	Defines the desired limit of REGISTER refreshes from all the UAs. Each second of time, sipd counts the number of REGISTER/200-OK responses sent back. When the threshold is exceeded, it increments the expire time (based on NAT interval) by one second and resets the count. By default no threshold is applied. The recommended value is somewhat dependent on the Net-Net SBC hardware used, but 300 can be used as an initial value.
<code>max-routes=<number of routes></code>	Restricts the number of routes through which the sipd will iterate from a local policy lookup. For example, setting this option to 1 causes the Net-Net SBC to only try the first, best, route. Setting this option to 0, or omitting it, lets the Net-Net SBC use all of the routes available to it (with the priority scheme for route matching). When you test a policy using the test-policy CLI command, this option is not recognized and all options that match the criteria are displayed.
<code>max-udp-length=0</code>	Setting this option to zero (0) forces sipd to send fragmented UDP packets. Using this option, you override the default value of the maximum UDP datagram size (1500 bytes; sipd requires the use of SIP/TCP at 1300 bytes).

Option	Description
<code>media-release=<header-name>[;<header-param>]</code>	Enables the multi-system media release feature that encodes IP address and port information for the media streams described by SDP. It lets another Net-Net SBC decode the data to restore the original SDP, which allows the media to flow directly between endpoints in the same network (that is serviced by multiple Net-Net SBCs). The media release information can appear in the following places: • SIP header P-Media-Release: <encoded-media-interface-information> • Header parameter on a SIP header Contact: <sip:1234@abc.com> ; acme-media=<encoded-media-interface-information> • SDP attribute in the message body a=acme-media: <encoded-media-interface-information> Option includes the following: • <header-name> is SIP header in which to put the information or the special value sdp, which indicates the information should be put into the SDP. • <header-param> is the header parameter name in which to put the information or in the case of the special header name value sdp, it is the SDP attribute name in which to put the information. They identify to where the encoded information is passed. If you do not specify a header, P-Media-Release is used.
<code>refer-to-uri-prefix=<prefix></code>	Defines a prefix to be matched against the userinfo part of Contact headers (config=), of which the Net-Net SBC should create a B2BUA map. This ensures that outgoing messages include the correct userinfo value. This option is used to enable add-on conferencing.
<code>reg-cache-mode=<mode></code>	Affects how the userinfo part of Contact address is constructed with registration caching. <mode> values are: • none: userinfo from the received (post NAT) Contact is retained • from: userinfo from the From header is copied to the userinfo of the forwarded Contact header • append: append the UA's Contact address into a cookie appended to the userinfo from the original Contact userinfo. For HNT, the NAT/firewall address is used. • append-from: takes userinfo from the From header and appends the encrypted address to the userinfo from the original Contact userinfo. For HNT, the NAT/firewall address is used. The from mode is used with softswitches that do not use the cookies used by the Net-Net SBC. It also helps limit the number of bytes in the userinfo; which might create duplicate contacts. For example, if the Net-Net SBC's address is 1.2.3.4, both 1234@5.6.7.8 and 1234@4.3.2.1 will result in a Net-Net SBC contact of 1234@5.6.7.8.

Option	Description
reg-contact-user-random	Enables the Net-Net SBC to treat endpoints that randomly change their contact user name every time they re-register as a registration refresh, rather than forward every re-registration to the registrar with the new contact information.
register-grace-timer	Makes the grace time for the SIP Registration configurable. You can configure the grace timer in seconds.
reinvite-trying=[yes]	Causes the Net-Net SBC to send a 100 Trying for re-INVITES, which is normally suppressed. If you enter the option name but omit the value yes, the option is still active.
reject-interval=<value>	Acts as a multiplier to increase the value presented to the UAC in the Retry-After field. For example, if reject-interval=5 (reject interval is set to 10); at a 90% rejection rate the Net-Net SBC sends "Retry-After: 45". When rejecting calls because of CPU load limiting, the Net-Net SBC adds a "Retry-After" parameter to the error response (typically 503 Service Unavailable). By default the Net-Net SBC sets the Retry-After value to be 1/10th of the current rejection rate.
reject-register=[no refresh]	Allows REGISTER messages through even during load limiting. By default, REGISTER messages are subject to load limiting.
response-for-not-found=<response code>	Change the 404 Not Found generated by the Net-Net SBC to a different response code.
route-register-no-service-route	Prevents the use of the Service-Route procedure to route the subsequent Register requests. If this option is not configured, the Net-Net SBC routes the Register requests using the Service-Route procedure.
sdp-insert-sendrecv	With this option configured, the Net-Net SBC automatically inserts a "sendrecv" directional attribute to the media session when either the offerer or answerer does not disclose its directional attribute.
set-inv-exp-at-100-resp	Set Timer C when a 100 Trying response is received (instead of waiting until 1xx (> 100) is received). If the Net-Net SBC does not receive a 100 Trying response within Timer B, the call should be dropped because there is a problem communicating with the next hop.
strip-domain-suffix-route	Causes sipd to strip any Router headers from the inbound messages coming to the external address of a SIP NAT; if the message contains a FQDN that matches the configured domain suffix for that SIP NAT.

Option	Description
supress100rel	For IWF-originated calls, the supress100rel option suppresses the SIP 100rel option tag on a per-H.323 interface (stack) basis.
video-sbc-session	Use with drain-sendonly for conference floor support. When configured with drain-sendonly and when the Net-Net SBC receives an SDP, the Net-Net SBC proxies the m=control and its related a= and c= unchanged. Although media streams are allocated for this m line, an actual flow is not set up. SDP received with the following: m=video a=sendonly is sent out as the following: m=video a=sendonly a=X-SBC-Session

SIP Interface Options

The following table lists the SIP interface options supported by the Net-Net SBC.

Option	Description
contact-endpoint=<endpoint name>	The Net-Net SBC inserts the endpoint IP address and port into the Contact headers as messages egress using that SIP interface. The inserted data is the same as the information received in the Request or Response being forwarded. If the endpoint name is not specified, the default value endpoint is used.
contact-firewall=<firewall name>	The Net-Net SBC inserts the firewall IP address and port into the Contact headers as messages egress using that SIP interface. The inserted data is the same as the information received in the Request or Response being forwarded. If the endpoint name is not specified, the default value firewall is used.
contact-vlan=<VLAN/realm name>	The Net-Net SBC inserts the realm and VLAN ID into the Contact headers as messages egress using that SIP interface. The inserted data is the same as the information received in the Request or Response being forwarded. If the endpoint name is not specified, the default value vlan is used.
dropResponse=<400:403:503:3XX: 4XX>	Allows the Net-Net SBC to drop specified responses according to their status codes. You configure this option either with specific status codes or with status code ranges. For example, you can enter the 400 or 403 response, or you can enter 3XX to include all 300 series status codes.
max-udp-length=0	Indicates the maximum UDP datagram size. (See entry for global SIP options for use globally). - If not configured globally or for a SIP interface, the default value of 1500 is used - If configured globally, that value is used for all SIP interfaces. - If configured for the SIP interface, the value is used only for that specific SIP interface.

Option	Description
reg-local-expires	Prevents the premature expiration of registration cache entries, which can occur when multiple Net-Net SBCs are provisioned with the same address of record. When you use registration caching and HNT, the Net-Net SBC adjusts the expiration time it sends to the user agents in REGISTER responses. It bases the time on the registration interval you configure.
reg-overload-protect	Provides SIP registration overload protection, which augments the Net-Net SBC's pre-existing protection methods by causing the Net-Net SBC to temporarily promote an endpoint to the trusted level when it receives the 401 or 407 response.
reg-via-key=<all: anonymous>	The Net-Net SBC checks whether the reg-via-key option is configured for the access-side SIP interface where a REGISTER is received. If enabled, the Net-Net SBC makes the via-key by adding the IP address from the Via header to the firewall address (if there is a firewall present between the Net-Net SBC and the endpoint). When an INVITE arrives at a SIP interface where this option is enabled, the Net-Net SBC determines whether the From header is anonymous or not. If it is anonymous, the Net-Net SBC uses the Via-key to find the registration entry.
response-for-not-found=<response code>	Changes the 404 Not Found generated by the Net-Net SBC to a different response code.
strip-route-headers	Causes the Net-Net SBC to disregard and strip all route headers for requests received on a SIP interface.
udp-fallback	If used and a request exceeds the max-udp-length-200 value, the Net-Net first uses TCP as the transport protocol. If the SIP endpoint does not support TCP and a RST is received, the Net-Net DS fallbacks to UDP and tries the request again.
via-header-transparency	Enables the Net-Net SBC to insert its Via header on top of the top-most Via header received from user equipment (UE). It then forwards it on to the IP Multimedia Subsystem (IMS) core with the original Via header now located as the bottom-most Via header. The Net-Net SBC still replaces the Contact and other header addresses with its own, and does not pass on the core's Via headers in outbound requests.

SIP Session Agent Options

The following table lists the SIP session agent options supported by the Net-Net SBC.

Option	Description
dropResponse	The Net-Net SBC drops responses by specified status codes. The option value can contain one or more status codes separated by semicolons. Error ranges can also be entered. If any of the response codes matches then a response is not sent. If the dropResponse option is set in both the sip-interface and the session-agent elements, the session-agent setting takes precedence.
trans-timeouts=<value>	Defines the number of consecutive non-ping transaction timeouts that will cause a session agent to be put out of service. The default is 5 (the existing behavior). A value of 0 prevents the session agent from going out of service because of a non-ping transaction timeout.
via-origin=<parameter-name>	Causes a parameter to be included in the top Via header of requests sent to the session agent. The parameter indicates the source IP address of the corresponding request received by the Net-Net SBC. <parameter-name> defines the name of the parameter. If not specified, the default value origin is used.

SIP Realm Options

The following table lists the SIP realm options supported by the Net-Net SBC.

Option	Description
number-normalization	Applies to the SIP To URI. (Currently the Net-Net SBC supports number normalization on From and To addresses for both inbound and outbound call legs.) Number normalization includes add, delete, and replace string functions that result in consistent number formats. Number normalization occurs on ingress traffic, prior to the generation of accounting records or local policy lookups. (also applies for H.323 to SIP calls.)
preferred-codec	Reorders the default codec in an SDP offer to the preferred codec before it forwards the offer to the target endpoint by setting the preferred-codec=X (where X is the preferred codec) option in the realm and session agent configurations. You set it in the realm from which the Net-Net SBC receives SDP offers (in requests or responses), and for which the media format list needs to be reordered by the Net-Net SBC prior to being forwarded. To configure additional codec ordering support for cases when a response or request with an SDP offer is from a session agent, you can set this option in the session agent configuration. When you enable this feature, you increase the probability that the target endpoint will choose the preferred codec for its SDP answer, thereby avoiding use of the undesired codec.

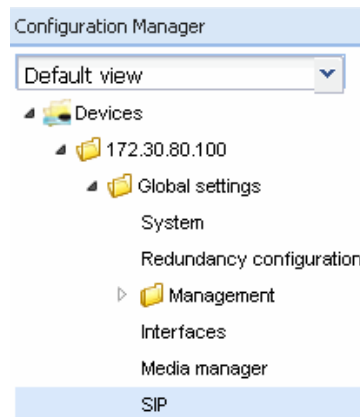
Configuring SIP Options

You access the different types of options from different windows in Net-Net Central. The following instructions tell you how to access this different types of options. Once you have accessed the options, the process of selecting them is the same for all types.

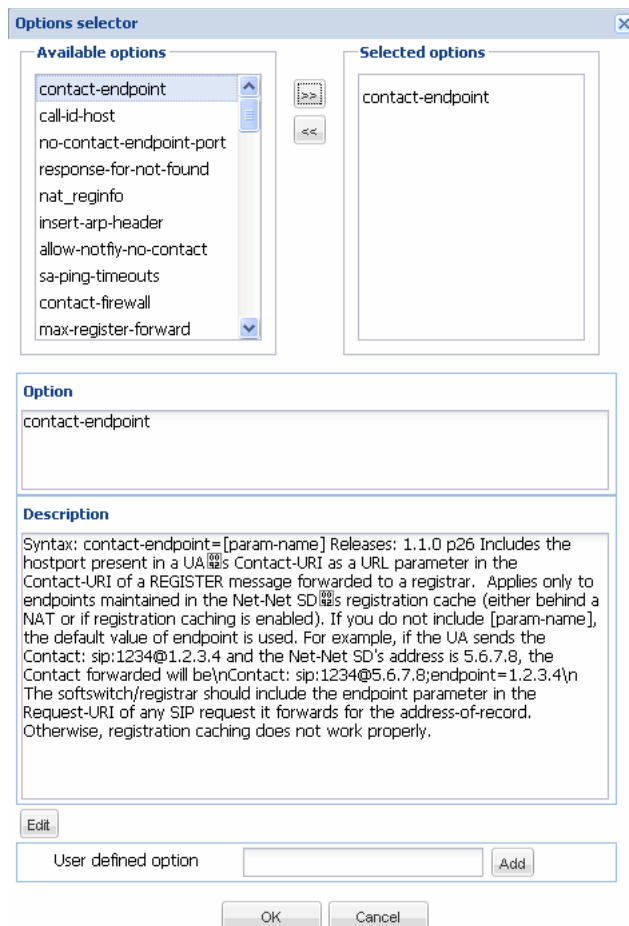
Accessing Global Options

To access global SIP options:

1. Expand the Configuration Manager slider > Global settings.



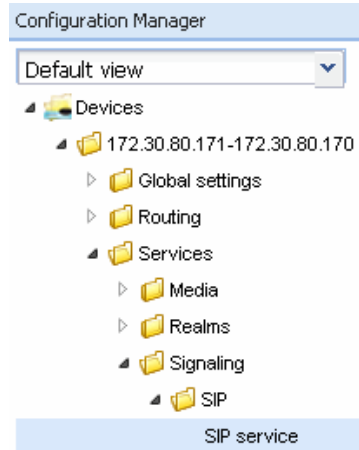
2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to the Option section of the content area.
5. Click **Options**. The Options dialog box appears.
6. Select options.



Accessing Interface Options

To access SIP interface options:

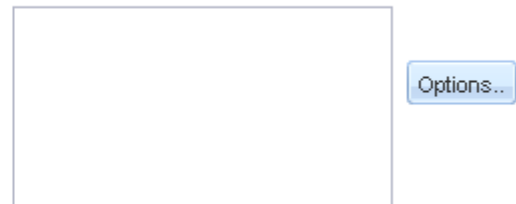
1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Select the SIP interface you want to configure in the table and click **Edit..**
4. Scroll to the Options in SIP service section of the content area.

Options in SIP service

Options

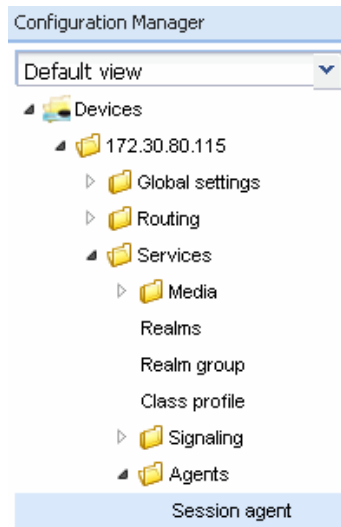


5. Click **Options**. The Options dialog box appears.
6. Select options.

Accessing Session Agent Options

To access session agent SIP options:

1. Expand the Configuration Manager slider > Services > Agents.

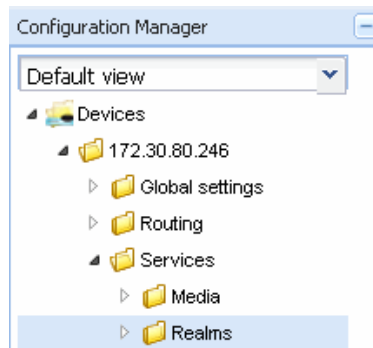


2. Click Session agent. The Session agent table appears in the content area.
3. Click the session agent you want to configure in the table and click **Edit**.
4. Scroll to the SIP options for this SA section.
5. Click **Options**. The Options Dialog box appears.
6. Select the options you want.

Accessing Realm Options

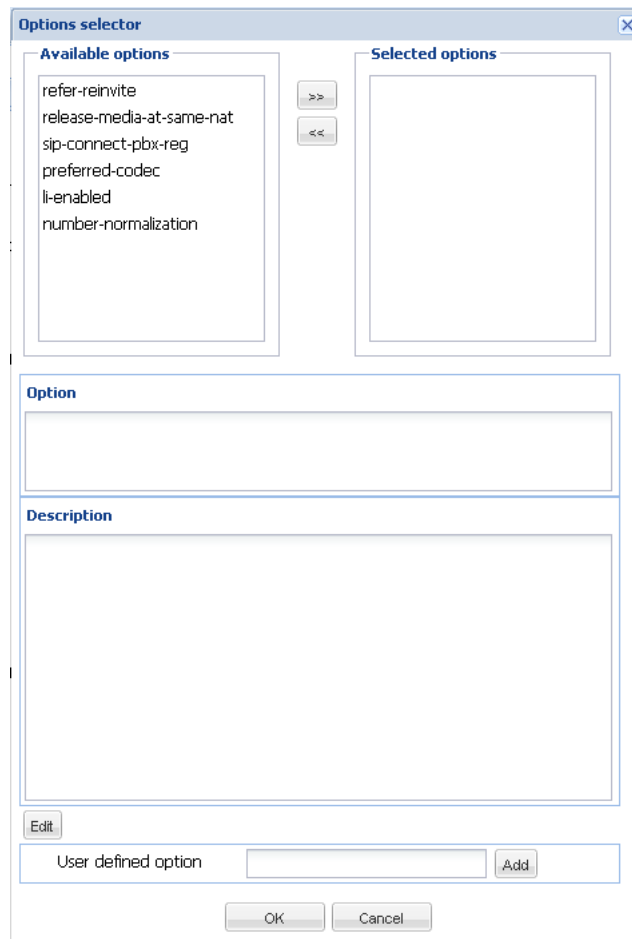
To access SIP realm options:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm to which you want to configure in the table and click **Edit**.
4. Scroll to the Options section of the content area.

5. Click **Options**. The Options dialog box appears..

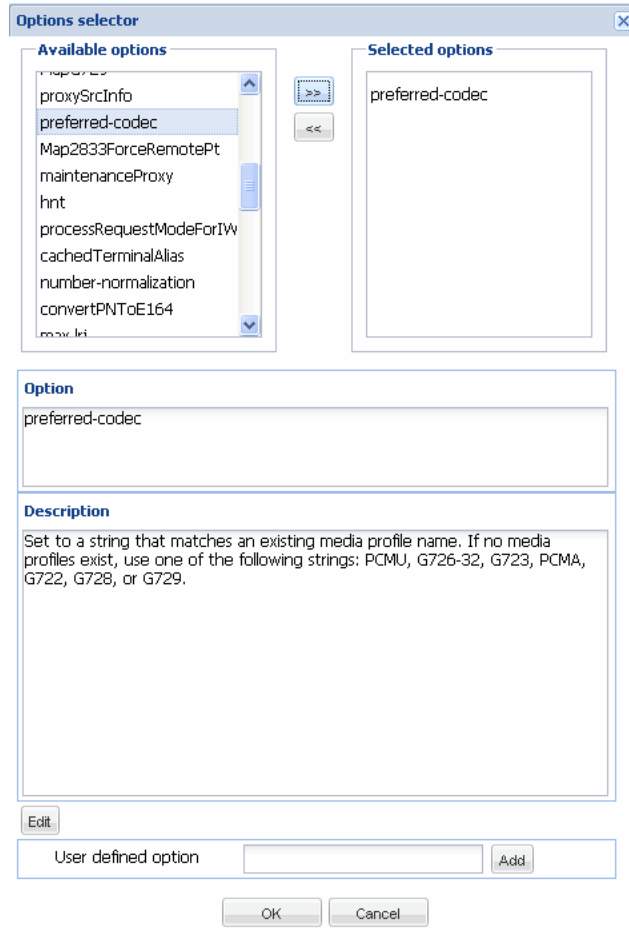


6. Select the options you want.

Selecting Options

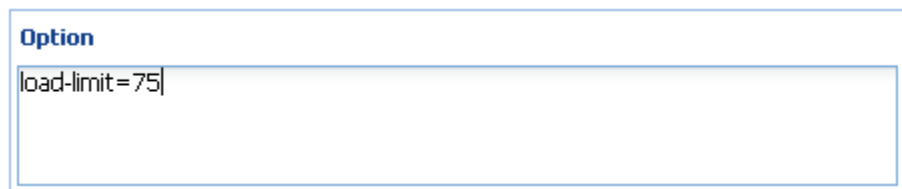
After you access the type of options you want, you follow the same procedure to select them in the Options Dialog box. The following instructions are for selecting a global SIP option.

1. Click the option name in the Select options scroll down list to select it.
2. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.



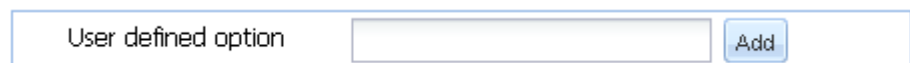
The **Options selector** dialog box is shown. It has two main panes: **Available options** and **Selected options**. In the **Available options** list, **preferred-codec** is selected. Between the panes are two buttons: a right arrow (>>) and a left arrow (<<). The **Selected options** pane contains **preferred-codec**. Below the panes, there is a section for the selected option. It has a label **Option** with the text **preferred-codec** and a label **Description** with the text: "Set to a string that matches an existing media profile name. If no media profiles exist, use one of the following strings: PCMU, G726-32, G723, PCMA, G722, G728, or G729." Below this is an **Edit** button. At the bottom, there is a text box labeled **User defined option**, an **Add** button, and **OK** and **Cancel** buttons.

3. In the option name text box, enter any values needed (if any). For example:



A text box labeled **Option** is shown. It contains the text **load-limit=75**.

4. Repeat steps 1 through 3 to select multiple options.
5. **User defined options**—Type the option name in the User defined options textbox to enter an option you have defined. For example:



A text box labeled **User defined option** is shown. It contains an empty text box and an **Add** button.

6. Click **Add**.

7. Click **OK** to close the Options dialog box.
8. Click **Apply** to complete entering SIP options.

Security

This section provides an overview of Net-Net SBC's security capability. Net-Net SBC security is designed to provide security for VoIP and other multi-media services. It includes access control, DoS attack, and overload protection, which help secure service and protect the network infrastructure (including the Net-Net SBC). In addition, Net-Net SBC security lets legitimate users to still place call during attack conditions; protecting the service itself.

Net-Net SBC security includes the Net-SAFE framework's numerous features and architecture designs. Net-SAFE is a requirements framework for the components required to provide protection for the Session Border Controller (SBC), the service provider's infrastructure equipment (proxies, gateways, call agents, application servers, and so on), and the service itself.

Denial of Service Protection

The Net-Net SBC Denial of Service (DoS) protection functionality protects softswitches and gateways with overload protection, dynamic and static access control, and trusted device classification and separation at Layers 3-5. The Net-Net SBC itself is protected from signaling and media overload, but more importantly the feature allows legitimate, trusted devices to continue receiving service even during an attack. DoS protection prevents the Net-Net SBC host processor from being overwhelmed by a targeted DoS attack from the following:

- IP packets from an untrusted source as defined by provisioned or dynamic ACLs
- IP packets for unsupported or disabled protocols
- Nonconforming/malformed (garbage) packets to signaling ports
- Volume-based attack (flood) of valid or invalid call requests, signaling messages, and so on.
- Overload of valid or invalid call requests from legitimate, trusted sources

Levels of DoS protection

The multi-level Net-Net SBC DoS protection consists of the following strategies:

- Fast path filtering/access control: involves access control for signaling packets destined for the Net-Net SBC host processor as well as media (RTP) packets. The Net-Net SBC accomplishes media filtering using the existing dynamic pinhole firewall capabilities. Fast path filtering packets destined for the host processor require the configuration and management of a trusted list and a deny list for each Net-Net SBC realm (although the actual devices can be dynamically trusted or denied by the Net-Net SBC based on configuration). You do not have to provision every endpoint/device on the Net-Net SBC, but instead retain the default values.
- Host path protection: includes flow classification, host path policing and unique signaling flow policing. Fast path filtering alone cannot protect the Net-Net SBC host processor from being overwhelmed by a malicious attack from a trusted source. The host path and individual signaling flows must be policed to ensure that a volume-based attack will not overwhelm the Net-Net SBC's normal call processing; and subsequently not overwhelm systems beyond it. The Net-Net SBC must classify each source based on its ability to pass certain criteria that is signaling- and application-dependent. At first each source is considered untrusted with the possibility of being promoted to fully trusted. The Net-Net SBC maintains two host paths, one for each class of traffic (trusted and untrusted), with different policing characteristics to ensure that fully trusted traffic always gets precedence.
- Host-based malicious source detection and isolation – dynamic deny list. Malicious sources can be automatically detected in real-time and denied in the fast path to block them from reaching the host processor.

Configuration Overview

NAT table entries are used to filter out undesired IP addresses (deny list). After the packet from an endpoint is accepted through NAT filtering, policing is implemented in the Traffic Manager based on the sender's IP address. NAT table entries are used to distinguish signaling packets coming in from different sources for policing purposes.

You can configure deny rules based on the following:

- ingress realm
- source IP address
- transport protocol (TCP/UDP)
- application protocol (SIP, MGCP)

You can configure guaranteed minimum bandwidth for trusted and untrusted signaling paths.

You can configure signaling path policing parameters for individual source addresses. Policing parameters include:

- peak data rate in bits per second
- average data rate in bits per second
- maximum burst size

Configuring Security

See the Security chapter in this guide for the configuration details.

SIP Unauthorized Endpoint Call Routing

The Net-Net SBC can route new dialog-creating SIP INVITEs from unauthorized endpoints to a session agent or session agent group; then rejection can occur based on the allow-anonymous setting for the SIP port. This type of provisional acceptance and subsequent rejection applies only to INVITEs; the Net-Net SBC continues to reject all other requests, such as SUBSCRIBE.

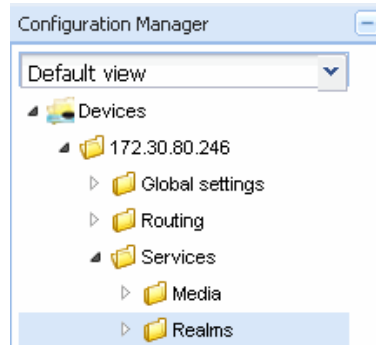
You might enable this feature if you have a network in which unauthorized SIP endpoints continually try to register even if the Net-Net SBC has previously rejected them and never will accept them. For instance, the user account associated with the endpoint might have been removed or core registrars might be overloaded.

Configuring SIP Unauthorized Endpoint Call Routing

You enable the routing of unauthorized endpoints to session agents and session agent groups that will reject them in the SIP interface configuration.

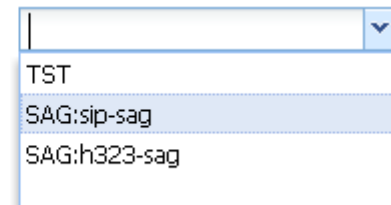
To enable SIP unauthorized endpoint call routing:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the Services and Protocols tab.
4. Select a SIP interface in the table and click **Edit**.
5. Scroll to SA or SAG to route unauthorized calls.
6. **SA or SAG to route unauthorized calls**—Click the name of the session agent or session agent group from the combination box to which you want calls from unauthorized endpoints routed. This parameter is blank by default, meaning the SIP unauthorized call routing feature is disabled.

SA or SAG to route unauthorized calls



7. Click **Apply**.

SIP NAT Function

This section explains how to configure the optional SIP NAT function. You can configure the SIP NAT function if you need to translate IP address and UDP/TCP port information. The SIP NAT function also prevents private IP addresses in SIP message URIs from traveling through an untrusted network.

Overview

The Net-Net system is an intermediary device that provides NAT functions between two or more realms. It translates IP addresses between untrusted and trusted networks using NAT. A trusted network is inside the NAT, and a untrusted network is outside the NAT. A NAT also lets a single IP address represent a group of computers.

For SIP, the SIP NAT function on the Net-Net SBC does the following:

- Routes SIP packets between the Net-Net system's SIP proxy (B2BUA) and external networks (or realms), including the translation of IP address and UDP/TCP port information.
- Prevents private IP addresses in SIP message URIs from traveling through the untrusted network. SIP NAT either translates the private address to one appropriate for an untrusted address or encrypts the private address into the URI.

Packets arriving on the external address (at port 5060) are forwarded to the Net-Net system's SIP proxy with the source address changed to the home address (at port 5060). When the Net-Net system's SIP proxy sends packets to the home address (at port 5060), they are forwarded to the external proxy address (and external proxy port), with the source address changed to the external address (at port 5060).

Note: The SIP config's NAT mode parameter works in conjunction with the SIP NAT function configuration. It identifies the type of realm in which the SIP proxy is located (public or private) and affects whether IPvr addresses in SIP messages are encoded.

The translation of URIs in the actual SIP message occurs as messages are received and sent from the Net-Net system's SIP proxy. For the messages being sent to the external network, the contents of the SIP message are examined after the translation to determine if the destination needs to be changed from the external proxy address to an address and port indicated by the SIP message. This process takes place so the request is sent to where the Request-URI or the Route header indicates, or so the response is sent to where the Via indicates.

NAT Modes

The specific addresses used in translating URIs in the SIP message depend on whether the Net-Net system is performing NAT functions for a trusted or untrusted network. This condition is determined by the NAT mode value you enter when you configure the SIP config element. The NAT modes are:

- **untrusted:** the SIP proxy is associated with an address for an untrusted network (the address value you entered when you configured the SIP interface's SIP port parameter), and the home address in the SIP NAT is the address of the external realm/network. When the URI contains the external address, it is translated to the SIP NAT's home proxy address (or to the SIP port address if the home proxy address field is empty). When a URI contains the external proxy address, it is translated to the home address.

If the URI contains any other private address (matching the realm's address prefix, identified in the SIP NAT's realm ID), it is encrypted and the address is replaced with the home address value. If the URI contains a user part, a suffix consisting of the user NAT tag and the encrypted address is appended to the user part. For example, with a user NAT tag value of `-private-`, the private URI of `sip@123192.169.200.17:5060` will become the public URI of `sip:123-private-eolmhet2chbl3@172.16.0.15`.

If there is no user part, the host consists of the host NAT tag followed by the encrypted address and the domain suffix. A `maddr` parameter equal to the home address (or received in the case of a Via header) is added to the URI. For example, with a host NAT tag value of `PRIVATE-` and a domain suffix value of `private.com`, the private URI of `sip:192.168.200.17:5060` will become the public URI of `sip:PRIVATE-eolmhet2chbl3.private.com:5060;maddr=172.16.0.15`.

- **trusted:** the SIP proxy is on a trusted network (the address value you entered when you configured the SIP interface's SIP port parameter), and the SIP NAT's external address is the public address of the external realm/network. When the URI contains the home address value, it is translated to the value set for the external proxy address. When the URI contains the SIP proxy's address, it is translated to the external address. If the URI contains any other private address (matching the realm's address prefix, identified in the SIP NAT's realm ID), the private address is encrypted and the address is replaced with the external address.

Note: Do not use the home proxy address value with private NAT functioning.

Adding a maddr Parameter to a URI

When you configure a SIP interface, you can configure the contact mode. The contact mode sets the contact header routing mode, which determines how the contact address from a trusted network is formatted. You set the contact mode to add a `maddr` parameter equal to the SIP proxy to the URI in the Contact header. For example, the URI from the prior example (`sip:192.168.200.17:5060`) becomes `sip:123-trusted-eolmhet2chbl3@172.16.0.15;maddr=172.16.0.12`.

Note: For SIP elements that do not support the `maddr` parameter, configure a Contact mode as `none`.

You might require this encryption to cause other SIP elements in the untrusted network to send requests directly to the SIP proxy. Otherwise, the requests are sent to the home address. However, responses sent by the SIP proxy will have the SIP proxy's source address, rather than the home address. Some SIP elements might drop responses that come from a IP address different from the one to which the request is sent.

About Headers

You can specify which SIP headers you want effected by the SIP NAT function. The URIs in these headers are translated and encrypted, the encryption occurs according to the rules of this SIP NAT function.

You can enter header values by using either the full header name or its corresponding abbreviation, if applicable. The following table lists the available headers and their corresponding abbreviations

Header	Abbreviation
Call-ID	i
Contact	m
From	f
Record-Route	none
Route	none
Ready-To	none
Replaces	none
Refer-To	r
To	t
Via	v

SIP sessions are terminated and re-originated as new sessions as they are routed through the Net-Net SBC. Among the actions performed, SIP headers are modified to prevent the transmission of IP address and route information.

Replacing Headers

In the SIP signaling message, any Via headers are stripped out and a new one is constructed with the Net-Net SBC's IP address in the sent-by portion. If a Contact header is present, it is replaced with one that has the Net-Net SBC's IP address. All other headers are subject to NATing based on the following rules:

- The Request-URI is replaced with the next hop's IP or FQDN address.
- All other headers are replaced based on the two SIP NAT function SIP NAT function rules

Mapping FQDNs

The Net-Net SBC maps FQDNs that appear in the certain headers of incoming SIP messages to the IP address that the Net-Net SBC inserts in outgoing SIP contact headers. The mapped FQDNs are restored in the SIP headers in messages that are sent back to the originator.

This feature is useful to carriers that use IP addresses in the SIP From address to create trunk groups in a softswitch for routing purposes. When the carrier's peer uses FQDNs, the carrier is forced to create trunk groups for each possible FQDN that it might receive from a given peer. Similarly, this can apply to SIP Contact and P-Asserted-Identity headers.

SIP NAT Function Cookies

Cookies are inserted to hide that information is coming from a realm external to the home realm. They are used when information needs to be placed into a given element of a SIP message that must also be seen in subsequent SIP messages within a flow. When forwarding a SIP message, the Net-Net SBC encodes various information in the outgoing message, which is passed from one side to another in SIP transactions.

SIP NAT function cookies let the Net-Net SBC hide headers, IP addresses, and SIP URIs. These cookies are included when certain conditions are present in Net-Net system SIP transactions.

Acme Packet's SIP NAT function cookies can be used in the userinfo, host, URL parameter, and *tel* URL parameter portions of the SIP message.

userinfo

The Net-Net system places a cookie in the userinfo portion of a SIP URI when a SIP header contains a SIP URI, and includes that header type in the list of headers to be hidden (encrypted) in the associated SIP NAT function. The cookie for the userinfo portion is the following:

```
[user nat tag][encrypted 13-byte host IP][encrypted 13 byte maddr IP (if present)]
```

where:

- [user nat tag] refers to the SIP NAT function's original user NAT tag field.
- [encrypted 13-byte host IP] refers to the host IP encryption.
- [encrypted 13 byte maddr IP (if present)] refers to the maddr IP encryption, if it exists.

With a user NAT tag of -acme, the following SIP-URI:

```
sip:6175551212@192.168.1.100
```

might be translated into:

```
sip:6175551212-acme-pfi1s7n2pstna@172.16.1.10
```

Note: Multiple additional cookies might be appended with each hop (for example, from the external proxy to the home proxy and back).

host

When hiding IP addresses in a SIP message, the SIP NAT function generates the following cookie for a SIP-URI with no userinfo portion:

```
[host nat tag][encrypted 13-byte host IP][encrypted 13 byte maddr IP (if present)][domain suffix]
```

where:

- [host nat tag] refers to the SIP NAT function's host NAT tag.
- [encrypted 13-byte host IP] refers to the host IP encryption.
- [encrypted 13 byte maddr IP (if present)] refers to the maddr IP encryption, if it exists.
- [domain suffix] refers to the SIP NAT function's domain suffix field.

With a SIP NAT function's host tag of **ACME-** and a domain suffix of **.acme.com**, the following SIP header:

Via: SIP/2.0/UDP 192.168.1.100:5060

might be translated into the following:

Via: SIP/2.0/UDP ACME-pf1s7n2pstna.acme.com

URL Parameter

If the SIP NAT function's use url parameter field has a value of **from-to** or **all**, the SIP NAT function places all cookies generated to hide SIP URIs in a custom tag appended to the header. Setting the use url parameter field to:

- **from-to** only affects the behavior of the SIP NAT function's cookies in the **From** and **To** headers.
- **all** affects all SIP headers processed by the SIP NAT function

The cookie is the following:

[;url-parameter]=[host nat tag][encrypted 13-byte host IP][encrypted 13-byte maddr IP]

where:

- **[;url-parameter]** refers to the SIP NAT function's parameter name field.
This cookie type is associated with the **all** and **from-to** field value options of the SIP NAT function's use url parameter field.
- **[host nat tag]** refers to the SIP NAT function's host NAT tag field.
- **[encrypted 13-byte host IP]** refers to the host IP encryption.
- **[encrypted 13 byte maddr IP (if present)]** refers to the maddr IP encryption, if it exists.

With a host NAT tag of **ACME-** and a parameter name of **acme_param**, the following SIP-URI:

sip:6175551212@192.168.1.100

might be translated into the following:

sip:6175551212@172.16.1.10;acme_param=ACME-pf1s7n2pstna.

tel URL

The SIP NAT function cookie is used when devices in your network are strict about the context portion of SIP messages regarding the conversion of tel URLs. This cookie for the tel URL parameter portion of a SIP message is the following:

"tel" URL parameter-[13-byte host IP][13 byte optional maddr IP]domain suffix

where:

- **tel URL parameter** refers to the SIP NAT function's use url parameter.
This cookie type is associated with the use url parameter's phone field value for the SIP NAT.
- **[13-byte host IP]** refers to the host IP encryption.
- **[13 byte optional maddr IP]** refers to the maddr IP encryption, if it exists.
- **domain suffix** refers to the SIP NAT function's domain suffix field.

Configuration Overview

Configuring the SIP NAT function falls into two areas, the SIP NAT interface parameters and the SIP NAT policies.

SIP NAT Interface

The following tables lists the SIP NAT function interface parameters you need to configure.

Parameter	Description
realm ID	Name of the external realm. The realm ID must be unique; no two SIP NATs can have the same realm ID. This realm ID must also correspond to a valid realm identifier entered when you configured the realm.
external proxy address	IP address of the SIP element (for example, a SIP proxy) in the external network with which the Net-Net system communicates. Entries must follow the IP address format.
external proxy port	UDP/TCP port of the SIP element (for example, a SIP proxy) in the external network with which the Net-Net system communicates. Minimum value is 1025, and maximum value is 65535. Default is 5060.
external address	IP address on the media interface in the external realm. Enter a value that ensures any packet with an external address value as its destination address is routed to the Net-Net system through the media interface connected to or routable from the external realm. Entries must follow the IP address format. To specify whether the external realm referenced in this field is private or public, configure the SIP config's NAT mode.
home address	IP address on the media interface in the home realm. Enter a value that ensures any packet with a home address value as its destination address must be routed to the Net-Net system through the media interface connected to or routable from the home realm. Entries must follow the IP address format. The value entered in this field must be different from the IP address value of the home realm's network interface element. The home realm network interface is associated with this SIP NAT by its realm ID and the realm's identifier and network interface value you entered when you configured the realm. The realm's network interface identifier value corresponds to this SIP NAT's realm ID, the SIP config's home realm ID, and the media manager's home realm ID.
home proxy address	Sets the IP address for the home proxy (from the perspective of the external realm). By default, this field is empty. An empty home proxy address field value signifies that there is no home proxy, and the external address will translate to the address of the Net-Net SBC's SIP proxy. Entries must follow the IP address format.

Parameter	Description
home proxy port	Sets the port number for the home realm proxy. Value can be set to zero (0). Minimum is 1025 and maximum is 65535. Default is 5060.
route home proxy	Whether to route all inbound requests for the SIP NAT to the home proxy. - enabled adds route if Request-URI is not the Net-Net SBC - disabled does not route inbound requests to the home proxy - forced always adds route

SIP NAT Function Policies

The following tables lists the SIP NAT function policy parameters you need to configure.

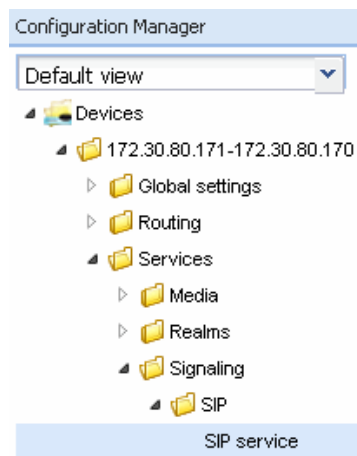
Parameter	Description
domain suffix	Domain name suffix of the external realm. The domain name suffix refers to and must conform to the hostname part of a URI. In combination with the user NAT tag and host NAT tag values, this value is used to help the Net-Net system identify an encoded URI that it needs to translate when moving between public and private realms. This suffix is appended to encoded hostnames that the SIP NAT function creates. For example, if the encoded hostname is ACME-abc123 and the domain-suffix value is .netnetsystem.com, the resulting FQDN will be ACME-abc123.netnetsystem.com.
address prefix	Defines which IP address prefixes from incoming messages require SIP-NAT encoding (regardless of the realm from which these messages came).
tunnel redirect	Controls whether Contact headers in a 3xx Response message received by the Net-Net SBC are NATed when sent to the initiator of the SIP INVITE message.
use url parameter	Establishes whether SIP headers will use the URL parameter entered in the parameter name for encoded addresses that the SIP NAT function creates. Also, if SIP headers will be used, which type of headers will use the URL parameter. For example, all headers or just the From and To headers. Enumeration field.
parameter name	Indicates the name of the URL parameter when use url applies. This field value will be used in SIP NAT encoding addresses that have a use url parameter value of either from-to or all.
user NAT tag	Identifies the prefix used when an address is encoded into the username portion of user@host;name=xxxx; where name = parameter name. The user NAT tag values can consist of any characters that are valid for the userinfo part of a URI. In combination with the domain suffix and host NAT tag field values, this value is used to help the Net-Net system identify an encoded URI that it needs to translate when moving between public and private realms.

Parameter	Description
host NAT tag	Identifies the prefix used when encoding an address into the hostname part of the URI or into a URL parameter. The host NAT tag values refer to domain labels and can consist of any characters that are valid for the hostname part of a URI. In combination with the domain suffix and user NAT tag values, this value is used to help the Net-Net system identify an encoded URI that it needs to translate when moving between public and private realms.
headers	Lists the SIP headers to be affected by the Net-Net system's SIP NAT function. The URIs in these headers will be translated and encrypted, and encryption will occur according to the rules of this SIP NAT.

Configuring SIP NAT Function

To configure SIP NAT function:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP NAT tab. The SIP NAT configuration table appears.
4. Click Add.
5. **Realm**—Select a Realm from the combination box for this SIP NAT configuration.
6. **Home realm addressing IP address**—Enter the IP address on the network interface in the home realm. This value must be such that any packet with a home address value as its destination address must be routed to the Net-Net system through the media interface connected to or routable from the home realm.

The value entered in this field **must be different** from the IP address value of the home realm's network interface element.

The home realm network interface is associated with this SIP NAT by its realm ID and the realm's identifier and network interface value you entered when you configured the realm. The realm's network interface identifier value corresponds to this SIP NAT's realm ID, the SIP configuration's home realm ID, and the media manager's home realm ID.
7. **SIP signaling interface IP address**—Enter the external IP address on the media interface in the external realm.

This value must be such that any packet with an external address value as its destination address is routed to the Net-Net system through the media interface connected to or routable from the external realm.

8. **SIP session agent IP address**—Enter the session agent's (proxy's) IP address in the external realm with which the Net-Net SBC communicates.

9. Select the SIP Nat configuration from the table and click **Edit**.
10. **SIP session agent IP port**—Enter the session agent's (proxy's) IP address's external proxy port value to identify the UDP/TCP port in the external realm which which the Net-Net SBC communicates. Values are:

- minimum: 1025
- maximum: 65535

SIP session agent IP port

5060

11. **Proxy IP address**—Enter the home proxy address to set the IP address for the home proxy (from the perspective of the external realm).

By default, this field is empty. No home proxy address entry signifies there is no home proxy, and the external address will translate to the address of the Net-Net system's SIP proxy.

Proxy IP address

127.90.120.65

12. **Proxy IP port**—Enter the home proxy port portion of IP address. Values are:

- empty
- minimum: 1025
- maximum: 65535

Proxy IP port

5060

13. **Proxy routing**—For the Proxy routing parameter, select the behavior of how to route inbound requests through the home proxy.

- **enabled**—All inbound requests for a specific SIP NAT are routed to the home proxy
- **disabled**—All inbound requests are not routed through the home proxy

- **forced**—The request is forwarded to the home proxy without using a local policy.

Proxy routing

SIP NAT Policies

1. **Realm IP address prefix override**—Enter the IP address prefix to match for this SIP NAT.

Realm IP address prefix override

2. **Return REDIRECT (3##) responses without NAT**—Select the behavior of NAT for certain headers in a 3xx Response message when sent to the initiator of the SIP INVITE message.

- **enabled**—Certain headers in a 3xx Response message are NATed.
- **disabled (default)**—Certain headers in a 3xx Response message are not NATed.

Return REDIRECT (3##) responses without NAT

3. **User name prefix**—Enter a value to identify the username prefix used for SIP URIs. The values you can use can include any characters valid for the userinfo part of a URI. This should be made unique for each realm and SIP NAT function.

User name prefix

In combination with the domain suffix and host NAT tag values, this value is used to help the Net-Net system identify an encoded URI that it needs to translate when moving between public and private realms.

4. **Host name prefix**—Enter a value for the host NAT tag field to identify the hostname prefix used for SIP URIs. The value refers to domain labels and can include any characters valid for the hostname part of the URI. This should be made unique for each realm and SIP NAT function.

Host name prefix

In combination with the domain suffix and user NAT tag values, this value is used to help the Net-Net system identify an encoded URI that it needs to translate when moving between public and private realms.

5. **Domain name suffix**—Enter the domain suffix of external realm. The domain name suffix refers to and must conform to the hostname part of a URI.

Domain name prefix

6. **Special name tag**—Select the behavior of how the Net-Net SBC uses the URL parameter for encoded addresses created by the SIP NAT function. If SIP headers will be used, this value identifies which types of headers will use the URL parameter. Values are:

- **none**: no headers will use the URL parameter for address encoding. The following example illustrates the functionality of a Net-Net system using a use url parameter value of none:

sip: 1234@1.2.3.4 is translated into **sip: 1234-acme-xxxx@5.6.7.8**

where -acme-xxxx is a cookie and xxxx is the encoded version of 1.2.3.4.

- **from-to:** From and To headers will use the URL parameter for address encoding. The following example illustrates the functionality of a Net-Net system using a use url parameter value of none:

sip: 1234@1.2.3.4 is translated into **sip: 1234@5.6.7.8; pn=acme-xxxx**

where -acme-xxxx is a cookie and xxxx is the encoded version of 1.2.3.4.

- **all:** all headers will use the URL parameter for address encoding. Acme Packet recommends not using this value because other SIP elements or implementations (other than the Net-Net SBC) might not retain the URL parameter in subsequent SIP messages that they send to the Net-Net SBC.
- **phone:**

If this field is set to either **from-to** or **all**, the Net-Net SBC puts the encoded address of the SIP NAT into a URL parameter instead of using the encoding name inside the userinfo part of the address.

Special name tag

7. **Parameter name**—Enter the hostname prefix to use if Special Name Tag combination box is set to from-to or all.

Parameter name

The parameter name value is used in SIP NAT encoding addresses that have the use url parameter values of from-to or all.

8. Click **Add** in the SIP NAT headers portion of the screen. The Add SIP NAT headers dialog box appears.
9. **SIP Header**—Enter the SIP header you want affected by the SIP NAT function in the SIP header field.

The URI in the header is translated, encrypted according to the SIP NAT function rule.

10. Click **Apply** to save the information and close the window.
11. Click **Apply**.

Realm Bridging

This section explains how to configure the internal routing among realms known as realm bridging. Realm bridging lets you cross-connect SIP interfaces. You can use one of the following two methods for bridging realms:

- local policy bridging: use this method to enable dynamic internal routing between realms if your SIP interfaces do not have the SIP NAT function applied.
- SIP NAT bridging: use this method if your SIP interfaces have the SIP NAT function applied.

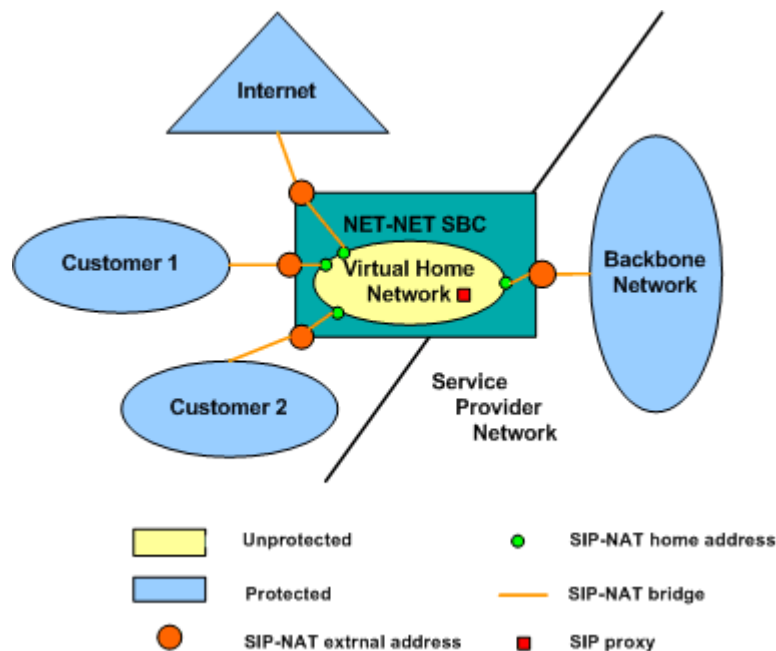
About SIP NAT Bridging

Each SIP NAT has a presence in two realms, trusted and untrusted. The SIP NAT bridge is the conduit for packages in and out of the home realm. It creates a bridge between realms by providing address translations; removing all references to the original IP addressing from the packets sent to the destination network.

With the SIP NAT bridge, an untrusted (or public) home network can reside within the Net-Net SBC, while the other entities (the backbone network, the Internet, or customer networks) are all trusted (or private). One of the primary functions of the SIP NAT bridge is to protect networks from one another so that address bases can remain hidden. Using a SIP NAT bridge, no one network has direct access to the data of other networks.

Establishing a SIP NAT bridge lets you route every SIP Request message through the backbone. Without using this functionality, it would appear as though all messages/sessions were coming from the Net-Net system's SIP proxy (the SIP server that receives SIP requests and forwards them on behalf of the requestor).

The following diagram illustrates this unprotected (or public) and protected (or private) division.



SIP NAT Bridge Configuration Scenarios

You can configure the SIP NAT bridge functionality in a many-to-one or a one-to-one relationship. For example, multiple customer SIP NATs can be tied to a single backbone SIP NAT, or a single customer SIP NAT can be tied to a single backbone SIP NAT.

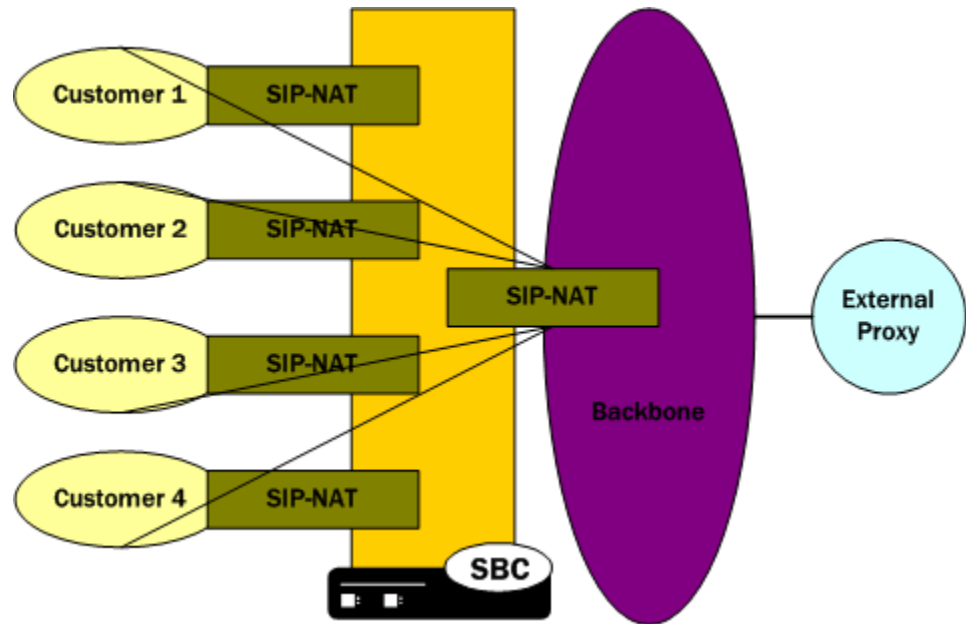
You might need to use several SIP NATs on the customer side while using only one on the backbone side in a many-to-one relationship. Or you might configure one

SIP NAT on the backbone side for every one that you configure on the customer side in a one-to-one relationship.

You can route all customer side SIP NAT requests to the corresponding backbone SIP NAT regardless of the Request URI. If a request arrives from the customer network with a Request URI that does not match the customer SIP NAT external address or the local policy that would route it to the backbone SIP NAT; the route home proxy value is used.

Many to One Configuration

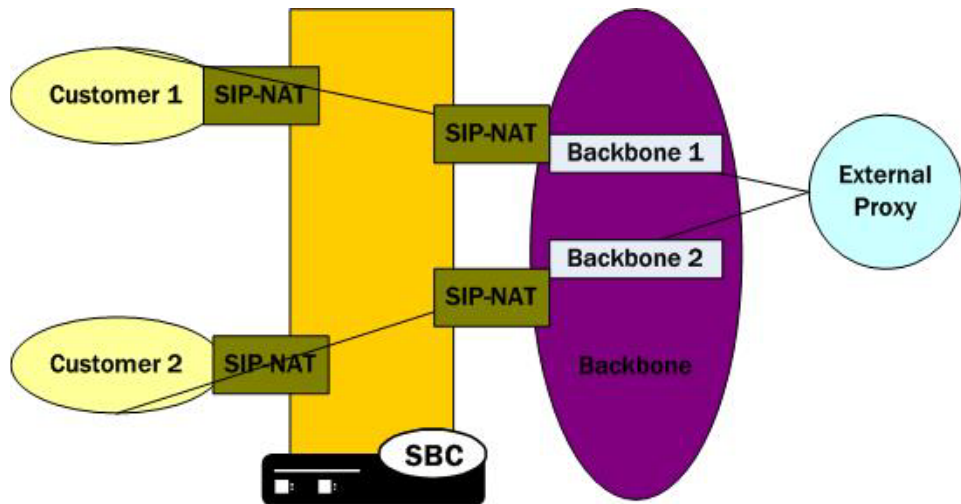
In the many-to-one scenario, multiple customer SIP NATs are tied to a single backbone SIP NAT. The following diagram illustrates the many-to-one SIP NAT bridge configuration.



One-to-One Configuration

In the one-to-one scenario, a single customer SIP NAT is tied to a single backbone SIP NAT. On the backbone SIP NAT side, you configure the home proxy address to match the home address of the customer SIP NAT. On the customer side, you configure the home proxy address to match the home address of the backbone SIP NAT.

The following diagram illustrates the one-to-one SIP-NAT bridge configuration.



SIP NAT Bridge Configuration

You create a bridge between SIP NATs by pointing them at one another. You point the SIP NATs at each other by configuring the home address and home proxy address to create the bridge. In addition, you can configure the route home proxy on the customer's side of a SIP NAT to force all requests to be routed to the corresponding backbone SIP NAT, regardless of the Request URI. You need to force requests when elements in the customer's network send requests with a Request URI that does not match the customer's SIP NAT external address. Or when the Request URI does not match a local policy element that would route the requests to the backbone SIP NAT.

You also need a home network to create a SIP NAT bridge. If you do not have a real home network, you need to create a virtual one. You also need to configure instances of the SIP NAT to create the SIP NAT bridge within your network.

Creating a Virtual Home Network

A virtual home network is a home network that resides entirely within the Net-Net SBC, as does a real home network. The difference between the two is the real home network also has a physical connection to the Net-Net SBC.

The internal home realm/network is usually configured with addresses within the special loopback range (127.0.0.0/8) as described in RFC 3330. This applies to the SIP port addresses for the home realm's SIP interface, and all home addresses for SIP NATs. The address 127.0.0.1 should not be used because it conflicts with the default loopback interface setup by the system for inter-process communication.

To create a virtual home network:

1. Set the name and subport ID of the network interface associated with the home realm element to lo0:0.
2. To enable the SIP proxy to listen for messages on the virtual home realm, configure the home realm ID. It must correspond to the realm's identifier, in which you set the network interface subelement to point to the appropriate network interface element.

The following table lists the field values you need to set when you are using SIP NAT bridge functionality and you do not have a real home network.

Configuration Element		Sample Values
realm configuration	identifier	home
	network interfaces	lo0:0
	address prefix	127.0.0.0/8
SIP configuration	home realm ID	home
	SIP ports address	127.0.0.100

Many-to-One Configuration

To configure many-to-one:

1. For the backbone SIP NAT, ensure the home proxy address field is blank.
2. For the customer side SIP NAT:
 - 2a. Set the home address to match the home address of the customer.
 - 2b. Set the home proxy address to match the backbone SIP NAT home address.
 - 2c. Set route home proxy to *forced*.

The following table lists the field values you need to set to create a many-to-one SIP NAT bridge.

SIP NAT Entity	Field	Sample Values
Backbone SIP NAT	home address	IP address of the home realm. For example: 127.0.0.120
	home proxy address	IP address of the home proxy from the perspective of the external realm. For a backbone SIP NAT, leave blank.
Customer SIP NAT	home address	127.0.0.120
	home proxy address	127.0.0.110
	route home proxy	forced

One-to-One Configuration

In the one-to-one scenario, a single customer SIP NAT is tied to a single backbone SIP NAT. The home proxy address field value of the backbone SIP NAT must match the home address of the customer SIP NAT. On the customer side, the home address of the customer SIP NAT should be defined as the home address of the customer, the home proxy address field value should match the home address of the backbone SIP NAT, and route home proxy should be set to *forced*.

The following table lists the field values you need to set to create a one-to-one SIP NAT bridge.

SIP NAT Entity	Field	Sample Values
Backbone SIP NAT	home address	IP address of the home realm. For example: 127.0.0.110
	home proxy address	IP address of the home proxy from the perspective of the external realm. 127.0.0.120
Customer SIP NAT	home address	127.0.0.120
	home proxy address	127.0.0.110
	route home proxy	forced

Shared Session Agent

Usually, the same set of servers (the external proxy) is used for all SIP NATs to the backbone network. In order to support redundant servers in the backbone of a SIP NAT bridge, the original egress realm as determined by the incoming Request URI needs to be retained after a local policy lookup.

When a request arrives at the Net-Net SBC, it determines the matching (target) session agent and, after the local policy is examined, sets the new outbound session agent to the one from the selected target.

If the target session agent's realm is set to *, the Net-Net SBC retains the original session agent's realm ID. Because the target session agent does not have a realm ID defined, the original egress realm is retained.

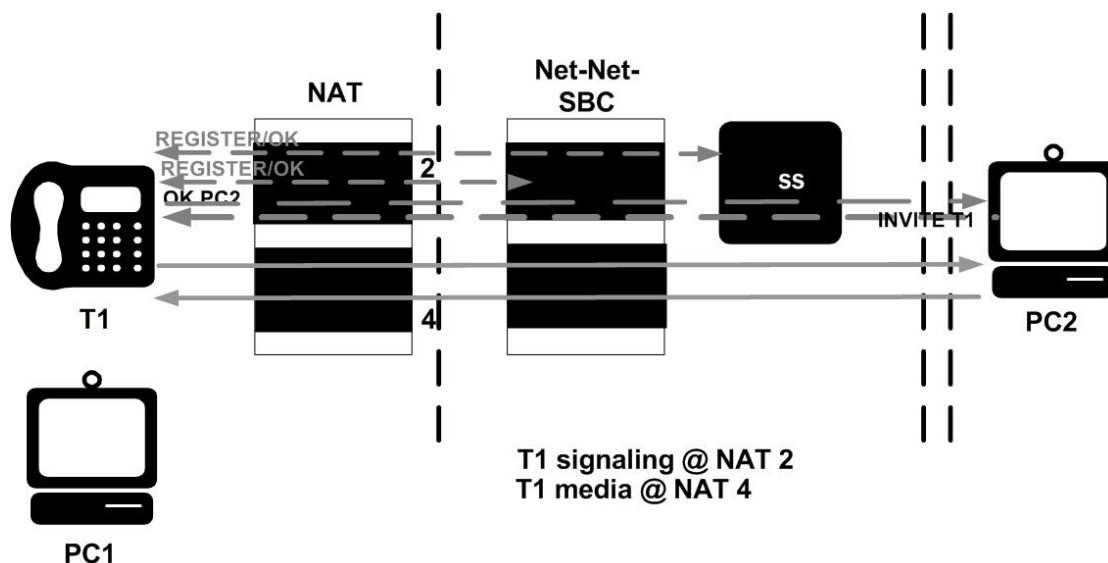
Hosted NAT Traversal

This section explains how to configure SIP Hosted Network Address Translation (HNT) traversal. SIP HNT lets endpoints (any system detected behind a NAT) behind a NAT/firewall device send and receive signaling and media using the Net-Net SBC as a relay.

About SIP HNT

SIP HNT is a technique the Net-Net SBC uses to provide persistent reachability for SIP UAs located in private Local Area Networks (LANs) behind Nat/firewall devices. It relies on frequent, persistent messaging to ensure that the binding on the intermediary NAT device is not torn down because of inactivity. HNT does not require support for the NAT in the SIP endpoint.

The following diagram illustrates SIP HNT traversal.



The Net-Net SBC's HNT function allows endpoints located behind NATs to communicate; providing means to traverse NATs. The Net-Net SBC interacts with endpoints (using SIP) to allow persistent inbound and outbound signaling and media communications through these NATs.

The Net-Net SBC automatically detects when an intermediate NAT exists between the UA and the Net-Net SBC by comparing the Layer 3 IP address of a REGISTER message with the IP address indicated within the UA. The Net-Net SBC sends signaling responses to the address and port that the request came from, rather than the address and port indicated in the request. The Via header in the request message indicates where the response should be sent.

Using HNT with Existing NAT Device

For network architectures in which premise devices and endpoints reside behind an existing NAT device, the Net-Net SBC's HNT function allows these premise NATs to be traversed without requiring an upgrade to the premise equipment, the deployment and management of additional premise-based hardware or software, or any NAT device configuration changes.

Registering Endpoints

The Net-Net SBC uses periodic endpoint registration messages to dynamically establish and maintain bindings in the NAT. These bindings keep a signaling port (port that is opened on a firewall to allow traffic to pass through it is a pinhole) open in the NAT that allows the inbound signaled communications to pass through. Using the endpoint registrations, the Net-Net SBC then maps the Layer 3 (OSI network layer that deals with switching and routing technologies for data transmission between network devices) IP address/port information from the NAT device to the Layer 5 (OSI session layer that deals with session and connection coordination between applications) entity (for example, user name or phone number) behind the NAT so that when an incoming signaling message is received, the Net-Net SBC sends it to the appropriate address and port on the NAT for the called party.

Establishing Media Flows

During call setup, the ports for bidirectional media flows are established dynamically. Since the media flows also pass through the Net-Net SBC, it can identify the IP address/port information on the NAT device used for the outgoing media coming from the user name/phone number. The Net-Net SBC then uses that same NAT's IP address/port information to send incoming media to the correct user name/phone number behind the NAT device.

Prerequisites

In order to achieve HNT, the endpoints involved must be capable of:

- symmetric signaling: sending and receiving SIP messages from the same transport address (IP address or User Datagram Protocol/Transmission Control Protocol (UDP/TCP) port)
- symmetric media: sending and receiving Real-Time Transport Protocol (RTP) messages from the same UDP port

These conditions are required to allow signaling and media packets back through the NAT (through the bound external address and port). These packets must come from the address and port to which the outbound packet that created the NAT binding was sent. The NAT sends these inbound packets to the source address and port of the original outbound packet.

When SIP HNT is used, the Net-Net SBC sends signaling responses to the address and port that the request came from rather than the address and port indicated in the request. The Via header in the request message indicates where the response should be sent.

Keeping the NAT Binding Open

Additional measures are also required to keep the NAT binding open because most NAT bindings are discarded after approximately a minute of inactivity. The Net-Net SBC keeps the SIP NAT binding open by returning a short expiration time in REGISTER responses that forces the endpoint to send frequent REGISTER requests.

In order to keep the NAT binding open for SIP, the Net-Net SBC maintains the registration state. When an endpoint first registers, the Net-Net SBC forwards that REGISTER message on to the real registrar. You can define the real registrar using either of the following methods:

- Configure the SIP config registrar host and registrar port to indicate the real registrar.

- Map the SIP config registrar host and registrar port values to the SIP NAT home proxy address and home proxy port values. Then configure the SIP NAT's external proxy address and external proxy port values to correspond to the real registrar.

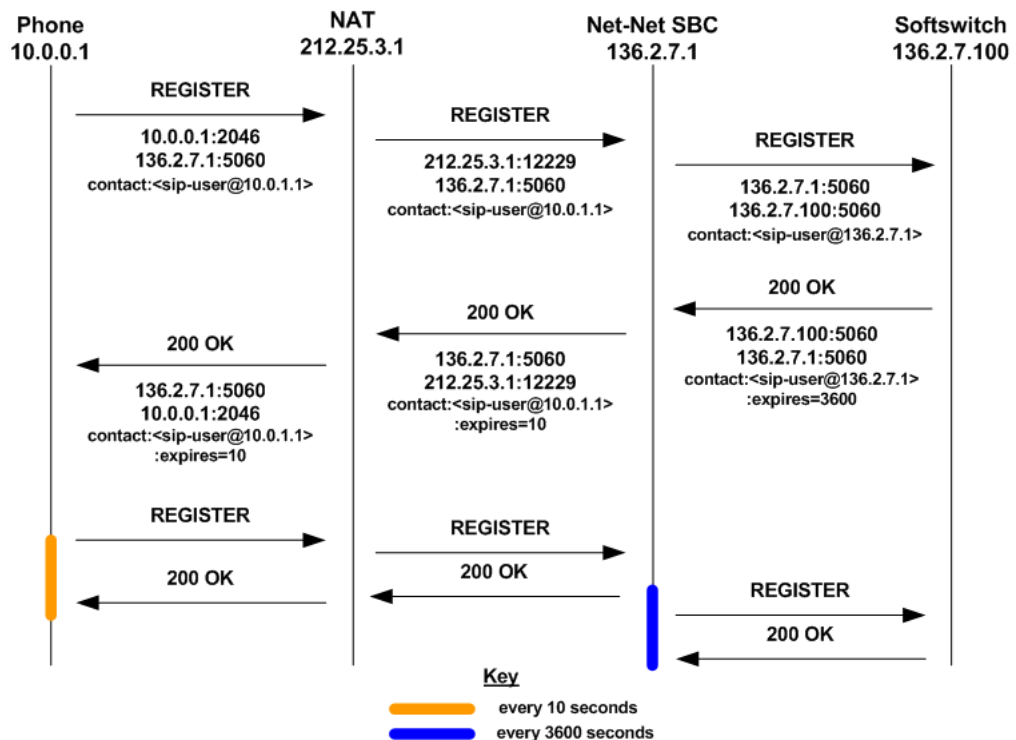
Note: A registrar can be located in a SIP NAT realm.

When a successful response is received, the Net-Net SBC caches the registration to memory. This cached registration lives for the length of time indicated by the expiration period defined in the REGISTER response message from the registrar. The response sent back to the endpoint has a shorter expiration time (defined by the SIP config's NAT interval) that causes the endpoint to send another REGISTER message within that interval. If the endpoint sends another REGISTER message before the cached registration expires, the Net-Net SBC responds directly to the endpoint. It does not forward the message to the real registrar.

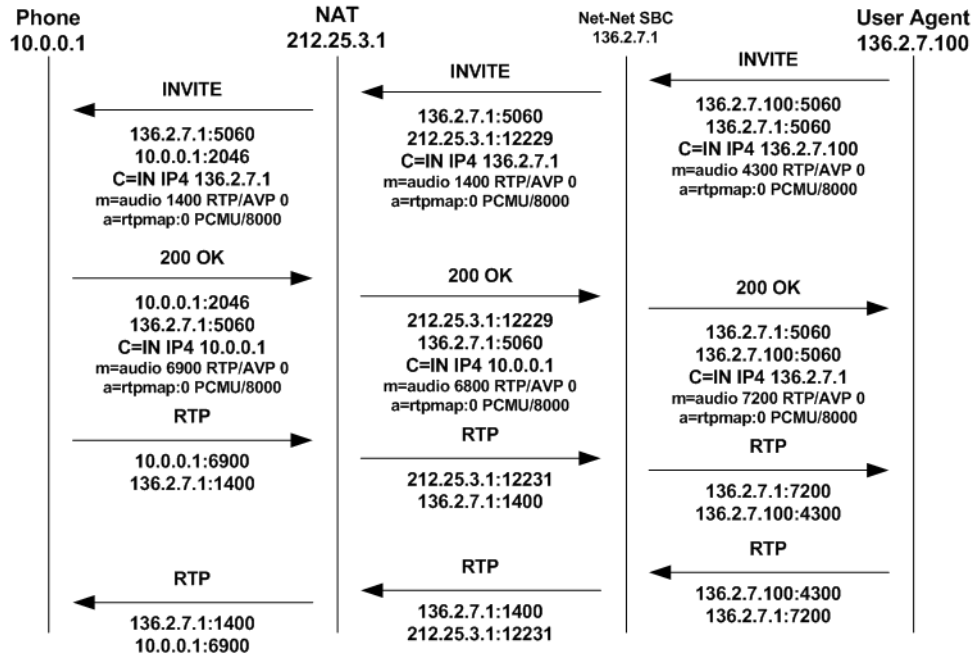
If the cached registration expires within the length of time indicated by the NAT interval, the REGISTER message is forwarded to the real registrar. If the Net-Net SBC does not receive another REGISTER message from the endpoint within the length of time indicated by the NAT interval, it discards the cached registration.

The Contact Uniform Resource Identifier (URI) in the REGISTER message sent to the registrar by the Net-Net SBC points at the Net-Net SBC so that the proxy associated with the real registrar sends inbound requests to the Net-Net SBC. This way, the inbound requests can be forwarded to the endpoint through the NAT binding.

The following example illustrates the SIP HNT registration call flow for the SIP HNT feature.



The following example illustrates the SIP HNT invitation call flow for the SIP HNT feature.



Working with Multiple Domains

You can use a wildcard (*) with the HNT feature to accommodate multiple domains and to allow the Net-Net SBC to cache all HNT endpoints. The wildcard functionality is enabled in the SIP config by entering an asterisk (*) in the registrar domain and registrar host fields.

The wildcard allows the use of either a local policy or Domain Name Service (DNS) to resolve the domain name to the correct registrar. Either method can be used to route the Fully Qualified Domain Name (FQDN) when you enter an asterisk (*) for the register host. An FQDN consists of an unlimited number of domain labels (domain names), each separated by a dot (.). The FQDN can include the top level domain name (for example, acmepacket.com).

In the hostname acme-packet.domainlbl.example100.com, the syntax is as follows:

- acme-packet is a domain label
- domainlbl is a domain label
- example100 is a domain label
- com is the top label

The information configured in a local policy is used before DNS is used. If the next hop destination address (defined in the local policy's next hop field) is an IP address, a DNS server is not needed. A DNS server is needed when the IP address of the next hop destination address is a FQDN or cannot be determined from the Net-Net SBC's configuration. Even with a configured local policy, the next hop destination address might be an FQDN that requires a DNS lookup.

If the registrar host does not use the wildcard, the Net-Net SBC always uses the configured address. You can limit the number of endpoints that receive the HNT

function. For example, you can use a non-wildcarded registrar domain field value (like `acme.com`) with a wildcarded registrar host field value.

HNT Configuration Overview

To configure SIP HNT NAT traversal, you need to configure both the SIP interface and the SIP config.

SIP HNT Single Domain Example

The following example shows values entered for the SIP config and SIP interface elements to configure SIP HNT for a single domain and registrar.

- SIP config

Parameter	Sample Value
registrar domain	netnetsystem.com
registrar host	192.168.12.1
registrar port	5060

- SIP interface

Parameter	Sample Value
NAT traversal	always
NAT interval	60
minimum registration expire	200
registration caching	disabled
route to registrar	enabled

SIP HNT Multiple Domain Example

The following example shows values entered for the SIP config and SIP interface elements to configure SIP HNT for a multiple domains and multiple registrars.

- SIP config

Parameter	Sample Value
registrar domain	*
registrar host	*
registrar port	0

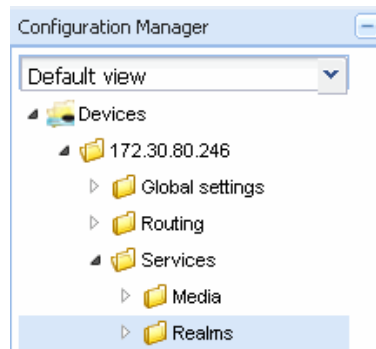
- SIP interface

Parameter	Sample Value
NAT traversal	always
NAT interval	60
minimum registration expire	200
registration caching	disabled
route to registrar	enabled

Configuring SIP Hosted NAT Traversal

This section describes how to configure SIP HNT .

1. Expand the Configuration Manager slider > Services.

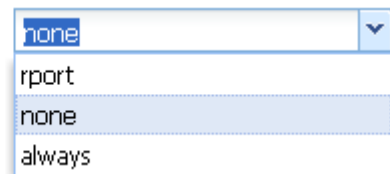


2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table.
4. Click **Edit**.
5. Click the Services and Protocols tab. The Realm services table appears.
6. Select a SIP interface service in the table and click **Edit**.
7. Click the Registration tab.
8. Scroll to the NAT traversal section.

NAT Traversal

1. **Endpoints type supported by NAT traversal**—Select the behavior of HNT for this SIP interface.
 - **none**—Disables the HNT feature for SIP (default value)
 - **rport**—SIP HNT function only applies to endpoints that include the rport parameter in the Via header and the sent-by of the topmost VIA matches the Contact-URI host address, both of which must be different from the received Layer 3 address.
 - **always**—SIP HNT applies to requests when the sent-by of the topmost VIA matches the Contact-URI host address, both of which must be different from the received Layer 3 address. (Even when the rport parameter is not present.)

Endpoints type supported by NAT traversal



2. **Endpoint to SD registration interval UDP (sec)**—Set the expiration time in seconds for the Net-Net SBC's cached registration entry for an HNT endpoint.
 - minimum: zero (0) seconds (default value)
 - maximum: 999999999 seconds

Acme Packet recommends setting the NAT interval to one-third of the NAT binding lifetime. A NAT binding lifetime is the network connection inactivity

timeout. The value is configured (or hardwired) in the NAT device (firewall). This timer is used to prevent the NAT device from keeping an unused port open.

Endpoint to SD registration interval UDP
(sec)

3. **Endpoint to SD registration interval TCP (sec)**—Set the registration cache expiration time in seconds for endpoints' behind a NAT device that register over TCP.

Endpoint to SD registration interval TCP
(sec)

4. **SD to registrar registration interval (sec)**—Set the minimum registration expiration time in seconds for endpoints' cached registration entries. The value you enter here sets the minimum registration expiration time in seconds for HNT registration caching.

- minimum: zero (0) seconds
- maximum: 999999999 seconds

This value defines the minimum expiration value the Net-Net SBC places in each REGISTER message it sends to the real registrar. In HNT, the Net-Net SBC caches the registration after receiving a response from the real registrar and sets the expiration time to the NAT interval value.

Some UAs might change the registration expiration value they use in subsequent requests to the value specified in this field. This change causes the Net-Net SBC to send frequent registrations on to the real registrar.

SD to registrar registration interval (sec)

Registration Handling

1. **Send transactions to registrar**—Enable routing to the registrar to send all requests that match a cached registration to the destination defined for the registrar host; used when the Request-URI matches the registrar host value or the registrar domain value, not the Net-Net SBC's address. Because the registrar host is the real registrar, it should send the requests back to the Net-Net SBC with the Net-Net SBC's address in the Request-URI.

For example, you should enable routing to the registrar if your network uses a Net-Net SBC and needs requests to go through its service proxy, which is defined in the registrar host field.

Send transactions to registrar

2. **Registration caching**—Enable it for use with all UAs, not just those that are behind NATs. By default, this field is set to disabled. If enabled, the Net-Net SBC caches the Contact header in the UA's REGISTER request when it is addressed to one of the following:
 - Net-Net SBC
 - registrar domain value
 - registrar host value

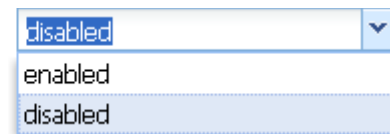
The Net-Net SBC then generates a Contact header with the Net-Net SBC's address as the host part of the URI and sends the REGISTER to the destination defined by the registrar host value.

Whether or not SIP HNT functionality is enabled affects the value of the user part of the URI sent in the Contact header:

- **HNT enabled**—The Net-Net SBC takes the user part of the URI in the From header of the request and appends a cookie to make the user unique. A cookie is information that the server stores on the client side of a client-server communication so that the information can be used in the future.
- **HNT disabled**—The user part of the Contact header is taken from the URI in the From header and no cookie is appended. This is the default behavior of the Net-Net SBC.

When the registrar receives a request that matches the address-of-record (the To header in the REGISTER message), it sends the matching request to the Net-Net SBC, which is the Contact address. Then, the Net-Net SBC forwards the request to the Contact-URI it cached from the original REGISTER message.

Registration caching



3. **SD to registrar registration interval (sec)**—Enter the Net-Net SBC's cached registration entry interval for a non-HNT endpoint in seconds. Enter the expiration time that you want the Net-Net SBC to use in the REGISTER response message sent back to the UA. The UA then refreshes its registration by sending another REGISTER message before that time expires.

- minimum: zero (0) seconds

A registration interval of zero causes the Net-Net SBC to pass back the expiration time set by and returned in the registration response from the registrar.

- maximum: 999999999 seconds

The default value is 3600 seconds (one hour). If the expiration time you set is less than the expiration time set by and returned from the real registrar, the Net-Net SBC responds to the refresh request directly rather than forwarding it to the registrar.

Note: With registration caching, there is no NAT; therefore, a short registration interval causes the UA to send excess REGISTER messages.

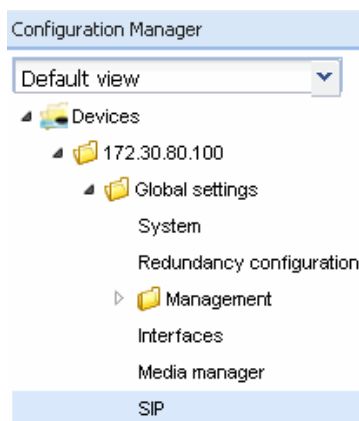
Although the registration interval applies to non-HNT registration cache entries, and the loosely related NAT interval applies to HNT registration cache entries, you can use the two in combination. Using a combination of the two means you can implement HNT and non-HNT architectures on the same Net-Net SBC. You can then define a longer interval time in the registration interval field to reduce the network traffic and load caused by excess REGISTER messages because there is no NAT binding to maintain.

SD to registrar registration interval (sec)

4. Click **Apply** to continue.

Configuring the Registrar

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click the Registration tab at the top of the content area.
5. Scroll to the Registrar section.
6. **Host name/IP address**—Enter the address of the registrar for which requests for registration caching, NAT traversal, and router to registrar options apply. You can use a specific hostname, a IP address, or a wildcard (*):
 - an asterisk (*) indicates normal routing (local policy, DNS resolution, and so on) is used to determine the registrar's address.
 - hostname: can consist of any alphanumeric character or any combination of alphanumeric characters (for example, `acme1.com`). The hostname can consist of any number of domain labels, separated by dots (.), and one top label. You can use the minimum field value of a single alphabetical character to indicate the top label value (for example, `c` to indicate `.com`).
 - IP address: must follow the dotted notation format. Each of the four segments can contain a numerical value between zero (0) and 255. For example, `192.168.201.2`. An example of an invalid segment value is 256. See the *ACLI User Interface* chapter of the *Net-Net ACLI Reference Guide* for more information about entries in IP address fields.
7. **Port**—Enter the SIP registrar port number. The SIP registrar server configured in this and the registrar host field is the real registrar. Or the values entered in those fields map to the home proxy address and home proxy port of the SIP NAT with external proxy address and external proxy port values that correspond to the real registrar.
 - minimum: zero (0) is the default
 - range: between 1025 and 65535
8. **Domain name**—Enter the domain to match against the host part of a URI to determine if a request is addressed to the registrar. If there is a match, the registration caching, NAT traversal, and route to registrar parameter values for the SIP interface are applied to the request. By default, this field remains empty.
 - an asterisk (*) to specify the values apply to all requests.
 - any alphanumeric character or any combination of alphanumeric characters. For example, `acme1.com`.

A hostname consists of any number of domain labels, separated by dots (.), and one top label. A top label is the last segment of the hostname. It must start with an alphabetical character. After the first character, a top label can consist of any number or combination of alphanumeric characters, including those separated by dashes. The dash must be preceded and followed by alphanumeric characters. A single alphabetical character is the minimum requirement for a hostname field (for example, c to indicate .com).

When the REGISTER message's Request-URI has an FQDN, it is matched against the registrar domain's value to determine if the message needs to be forwarded to the registrar port on the registrar host. The registrar domain's value is also used when route to registrar is set to enabled, to determine if a request needs to be forwarded to the registrar.

Only the right-hand part of the domain name in the Request-URI needs to match the registrar domain value. For example, `acme3.acmepacket.com` matches `acmepacket.com`. However, the entire domain label within the domain name must match. For example, the domain label "acme3.acmepacket.com" would not match "packet.com".

Registrar

Host name/IP address	*
Port	5060
Domain name	*

- Click **Apply** to complete configuring SIP HNT.

Adaptive HNT

This section explains how to configure adaptive HNT. The adaptive HNT expires feature allows the Net-Net SBC to automatically determine the maximum SIP REGISTER message expires time interval in order to keep each individual NAT pinhole open when performing SIP HNT.

Overview

Without adaptive HNT, the Net-Net SBC keeps NAT pinholes open and port mapping cached by forcing the UAC to send frequent SIP REGISTER messages. It does so by setting the expires time to a short interval. Some NATs only need a message to be sent by the private client once every twenty minutes, while other NATs delete their cache/pinhole in thirty seconds if no messages appear. Given this large variation in time intervals, the Net-Net SBC's nat-interval (expire time) has been set to a low value in order to support as many NAT types as possible. However, CPU performance and scalability issues result from such a small refresh time, especially when there is a very large number of potential registered users.

When you use adaptive HNT, the Net-Net SBC waits for a time interval and then sends a SIP OPTIONS message to the UAC to see if it can still be reached. If the UAC can still be reached, the Net-Net SBC increases the timer and tries again. In case the pinhole closes because it has exceeded the NAT's cache time, the Net-Net SBC sets the expires time to be slightly longer than the time it tests using the OPTIONS method. This way, the UAC will send another REGISTER message shortly thereafter and impact on service will be minimal.

Adaptive HNT Example

An example call flow using adaptive HNT involves a basic HNT user and a Net-Net SBC. It begins when the Net-Net SBC receives and forwards the 200 OK for the REGISTER message. Then the Net-Net sends an expires timer for slightly longer than the time for which to test; in this example, it begins the test for the amount of time set for the minimum NAT interval. It adds ten seconds to this time when it sends the expires timer. This way, there is time for the OPTIONS message to be sent before the REGISTER message is received (which would refresh the NAT's cache). The Net-Net SBC also tries to keep the REGISTER time short enough so that even if the NAT pinhole closes, there is minimal time before the UAC creates a new NAT binding by sending another REGISTER. Because a ten second interval may be too long, you might want to set this value to a better-suited time.

The test succeeds with a minimum test-timer because the UAC responded to the OPTIONS message. So the test-timer value is increased by thirty seconds and tried again. The expires time in the REGISTER message will be increased to the test-timer value plus ten seconds. This time, the UAC does not respond to the OPTIONS message even though it was sent multiple times. Because the OPTIONS fails, when the Net-Net SBC receives another REGISTER, it responds with the previously successful timer value (in this case, the minimum NAT interval).

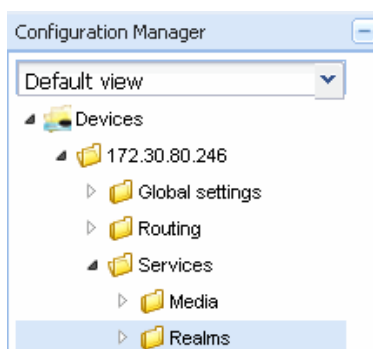
However, if the OPTIONS request succeeds, then the Net-Net SBC persists with the test until it fails or until the maximum NAT timer value is reached. In this case, when the OPTIONS message fails, the Net-Net SBC uses the last successful test-timer value as the time for the expires header in the 200 OK for the REGISTER message.

Configuring Adaptive HNT

To configure adaptive HNT:

You configure the SIP interface to set the state of this feature and to define the increments of time the Net-Net SBC uses to perform adaptive HNT. Remember that the Net-Net SBC uses the time you specify as the NAT interval, the supported time interval, as the basis on which to begin testing.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click Registration at the top of the content area.
7. Scroll to the Adaptive Hosted NAT Traversal section.

8. **Dynamic HNT**—Select enabled from the combination box to enable adaptive HNT.
9. **Maximum registration interval for SIP HNT (secs)**—Enter the amount of time in seconds that testing should not exceed for the maximum registration interval. The Net-Net SBC will keep the expires interval at this value. Retain the default value of 3600 or enter a new value.
10. **Interval increment for SIP HNT test (secs)**—Enter the amount of time in seconds to use as the increment in value in the SIP expires header. Retain the default value of 10 or enter a new value.
11. **Test increment value for SIP HNT test (secs)**—Enter the amount of time in seconds that will be added to the test timer. Retain the default value of 30 or enter a new value.

Adaptive Hosted Nat Traversal

Dynamic HNT	enabled
Maximum registration interval for SIP HNT (sec)	3600
Interval Increment for SIP HNT test (sec)	10
Test increment value for SIP HNT test (sec)	30

12. Click **Apply**.

Server Redundancy

This section explains how to configure SIP server redundancy. SIP server redundancy involves detecting that an upstream/downstream SIP signaling entity has failed, and adapting route policies dynamically to remove it as a potential destination.

Overview

You establish SIP server redundancy by creating session agents, which are virtual representations of the SIP signaling entities. These agents are then collected into a session agent group, which is a logical collection of two or more session agents that behaves as a single aggregate entity. For more information about session agents and session agent groups, see the Session Routing and Load Balancing chapter in this guide.

Rather than direct signaling messages to a single session agent (IP), the signaling message is directed to a session agent group (SAG). The group will have a set distribution pattern: hunt, round robin, proportionally distributed, and so on. Signaling is spread amongst the agents using this chosen pattern.

You direct the signaling message by configuring a route policy, known as a local policy, which determines where SIP REQUESTS should be routed and/or forwarded. The values in the To and From headers in the SIP REQUEST are matched with the content of the local policy within the constraints set by the session agent's previous hop value and SIP interface values such as the list of carriers.

To summarize, you need:

- two or more session agents
- a session group containing those session agents
- a local policy which directs traffic to the session agent group

Configuration Overview

You make a session agent group a target by using a local policy to select the next hop from the members of a session agent group. You need to set the replace URI field of the configured local policy to enabled; which causes NAT rules such as realm prefixing to be overridden. The replace URI field allows you to indicate whether the local policy's value is used to replace the Request-URI in outgoing requests. This boolean field can be set to either **enabled** or **disabled**.

When the SIP NAT's route home proxy field is set to **forced**, it forces the Request to be forwarded to the home proxy without using a local policy. When this option is set to either **disabled** or **enabled** and the Request-URI matches the external address of the SIP NAT, the local policy is used.

However, the local policy only replaces the Request-URI when the original Request-URI matches the Net-Net system's IP address or hostname. This behavior is in accordance with that described in RFC 3261. The original Request-URI will be the home proxy address value (the home address of the SIP NAT into the backbone) and not the Net-Net system's address.

Using strict routing, the Request-URI would be the next hop, but the message would also include a Route header with the original Request-URI. With loose routing, the Request-URI remains unchanged and the **next hop** value is added as the top Route header.

In some cases, the **next hop** field value must replace the Request-URI in the outgoing request, even if the original Request-URI is not the Net-Net system. To accomplish this, an option has been added to the local policy that causes the **next hop** value to be used as the Request-URI and prevents the addition of Route headers. This option is the **replace uri** value in the local policy.

The following table lists the policy attributes for the local policy:

Parameter	Description
next hop	IP address of your internal SIP proxy. This value corresponds to the IP address of the network interface associated with the SIP proxy.
realm	Number of the port associated with the SIP port.
replace uri	Stores the transport protocol used for sending an receiving signaling messages associated with the SIP port.
allow anonymous	Indicates whether this SIP port allows anonymous connections from session agents.

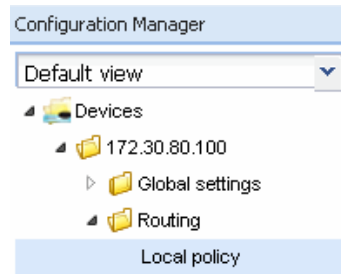
Note: You should also define the ping method intervals for the session agents so that the Net-Net SBC can detect when the agents are back in service after failure.

For more information about local policy, see the Session Routing and Load Balancing chapter in this guide.

Configuring SIP Server Redundancy

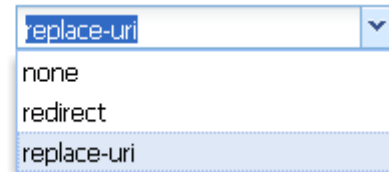
This section describes how to configure SIP server redundancy on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Click a policy name in the table and click **Edit**.
4. Scroll to the Route policy attributes table, select an attribute and click **Edit**.
5. **Action**—Select **replace-uri** from the Action combination box.

*Action



The replace URI field allows you to indicate whether the local policy's value is used to replace the Request-URI in outgoing requests.

6. Click **Apply** to close the Route policy window.
7. Click **Apply**.

Distributed Media Release

This section explains how to configure distributed media release (DMR). SIP DMR lets you choose whether to include multi-system (multiple Net-Net SBCs) media release information in SIP signaling requests sent to a specific realm.

Overview

The SIP DMR feature lets RTP/RTCP media be sent directly between SIP endpoints (for example, SIP phones or user agents) without going through a Net-Net SBC; even if the SIP signaling messages traverse multiple Net-Net SBCs. It encodes IP address and port information for the media streams described by the media, for example SDP.

With SIP DMR, the media realm and IP address and port information from the UA's SDP are encoded into SIP messages (either in the SIP header or in the SDP) as they enter the backbone network. The information is decoded by a Net-Net SBC from SIP messages that come from the backbone network. The decoded address and port information is put into the SDP sent the UAs in the access (private/customer) network.

This functionality lets the RTP/RTCP flow directly between the UAs in the access network without traversing the Net-Net SBCs and without passing into the

backbone network. The media can then flow directly between the two SIP endpoints in the same network, if it is serviced by multiple Net-Net SBCs.

You can enable this feature on a per-realm basis and multiple realms can be supported.

Endpoint Locations

You can configure the Net-Net SBC to release media when the source and destination of the call are in the same network, customer VPN, or customer LAN. In architectures that use DMR, the Net-Net SBC is only part of the media path for traffic that originates and terminates in different networks.

If configured to do so, the Net-Net SBC can release media:

- Between endpoints supported by a single Net-Net SBC
 - In the same network/VPN
 - In the same network behind the same NAT/firewall
- Between endpoints supported by multiple distributed Net-Net SBCs
 - In the same network/VPN

Location of the Encoded Information

Encoded media release information can appear in three different places:

- SDP attribute

Media release data can be encoded into an SDP attribute in the SIP message body (for example, `media-release=sdp;acme-media`). The encoded data is placed into an `acme-media` attribute in the SDP:

```
a=acme-media:<encoded-media-interface-info>
```

- SIP header parameter

Media release data can be placed in a header parameter of a SIP header (for example, `media-release=Contact;acme-media`). The encoded data is placed into an `acme-media` parameter in the Contact header:

```
Contact: <sip:1234@abc.com>;acme-media=<encoded-media-interface-info>
```

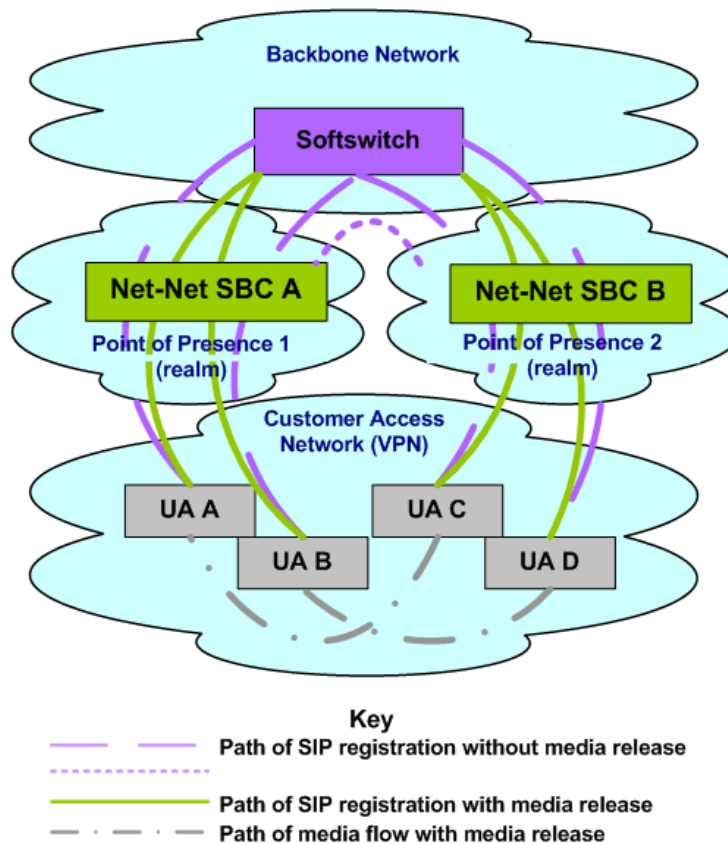
- SIP header

Media release data can appear in a SIP header (for example, `media-release=P-Media-Release`). The encoded data is placed into a `P-Media-Release` header:

```
P-Media-Release: <encoded-media-interface-info>
```

Example: Distributed Media Release

The following example shows the network diagram for DMR in a multiple-site VPN environment supported by multiple, distributed Net-Net SBCs.



As shown in the network diagram, UA A and UA B register with the softswitch through Net-Net SBC A while UA C and UA D register with the softswitch through Net-Net SBC B. Without DMR, the media for calls between UA A/UA B and UA C/UA D is steered through both Net-Net SBC A and Net-Net SBC B.

With SIP DMR, the media realm and IP address and port information from the UA's Session Description Protocol (SDP) is encoded into SIP messages (either in the SIP header or in the SDP) as they enter the backbone (public/service provider) network. The information is decoded from SIP messages that come from the backbone network. The decoded address and port information is put into the SDP sent to the UAs in the access (private/customer) network. This functionality allows for the RTP/RTCP to flow directly between the UAs in the access network without traversing the Net-Net SBCs and without passing into the backbone network.

Overview of SIP DMR Configuration

To configure SIP DMR:

1. Edit the SIP config element's option field.

The media-release="`<header-name>[:<header-param>]`" option defines how the SIP distributed media release feature encodes IP address and port information. If the media-release parameter is configured in the options field but no header is specified, the parameter value of `P=Media-Release` will be used. This parameter is optional and is not configured by default.

2. Enable SIP DMR for the entire realm by setting the realm config element's msm release field to enabled.

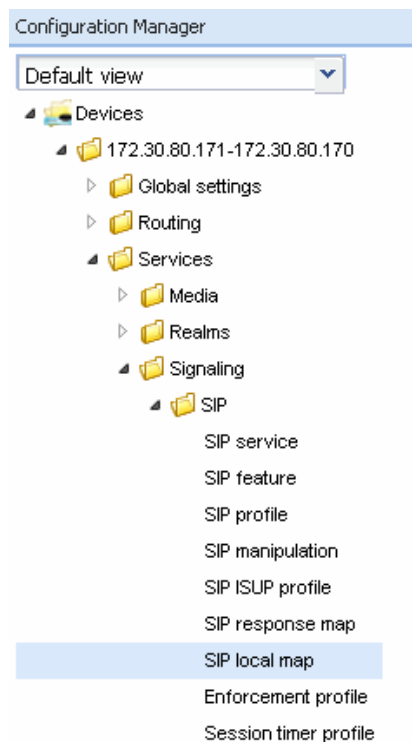
The media IP address and port information is encoded into outgoing SIP messages and decoded from incoming SIP messages for all of the realms (in each realm-config element) with which the SIP distributed media release will be used.

Note: You can also use the realm config element's mm in network field to release the media back to a connected network that has multiple realms. This field is not specific SIP distributed media release and it is not required for the SIP DMR to work. However, if this field is set to enabled and the ingress and egress realms are part of the same network interface, it lets the Net-Net SBC release the media.

Configuring Distributed Media Release

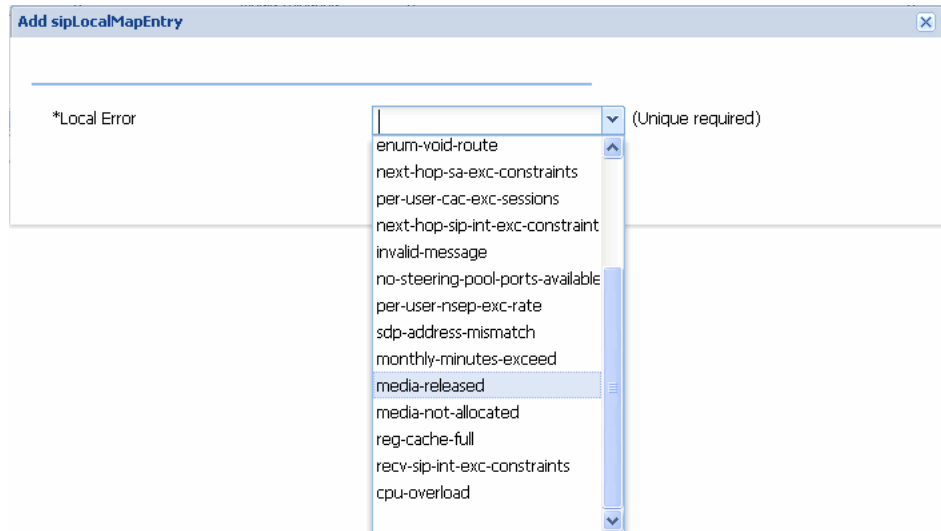
This section describes how to configure SIP distributed media release on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Services > Signaling > SIP.

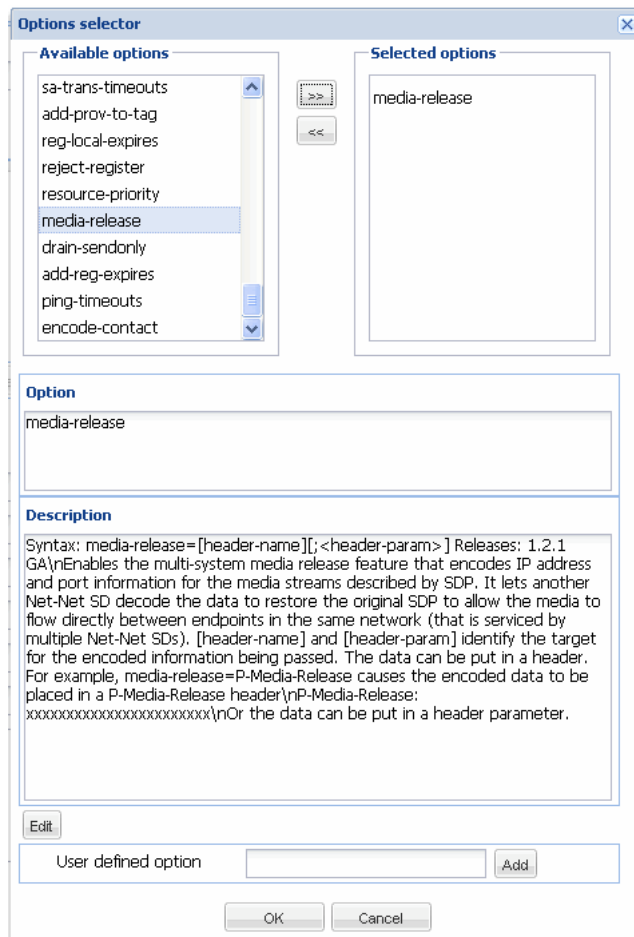


2. Click SIP local map.
3. Click **Add** to add the configuration. The SIP local map table displays in the content area.
4. Click Add.

5. Select media-released from the combination box.



6. Click **Apply**.
7. Click **Options**. The SIP option window appears.
8. Click  to move the option to the opposite list. The option name appears along with a description in the lower part of the window.



9. **media-release**—Edit the textbox to append one of the following:

- SDP attribute, for example:
media-release=sdp;acme-media
- SIP header parameter, for example:
media-release=sdp;acme-media
- SIP header, for example:
media-release=P-Media-Release

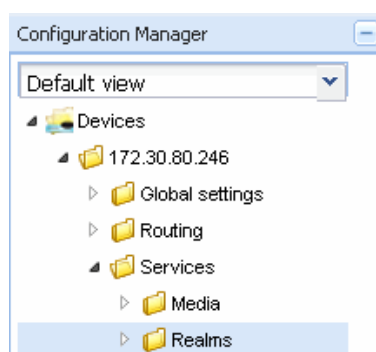
`media-release=P-Media-Release`

10. Click **OK** to save your edits and close the SIP options window.
11. Click **Apply** to continue.

Configuring the Realm Configuration

This section describes how to configure SIP distributed media release on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm and click Edit.
4. Scroll to Release media peer-to-peer between endpoints.
5. **Within this realm serviced by different Net-Net systems (SIP only)**—Select enabled from the combination box to enable DMR within this realm.

Manage media between endpoints

Within this realm serviced by this Net-Net system only	disabled
Within different realms in same subnet serviced by this Net-Net system only	enabled
Within this realm serviced by different Net-Net systems (SIP only)	enabled enabled disabled

6. Click **Apply** to complete configuring SIP distributed media release.

Add-On Conferencing

This section explains how to configure the add-on conferencing functionality. It also includes a description of the SIP B2BUA functionality related to the SIP add-on conferencing. This description includes information about Contact header mapping and processing and Refer-to header processing.

Overview

SIP add-on conferencing lets you:

- Use the Net-Net SBC's add-on conferencing feature for network architectures in which the conference initiator is located on a different network than that of the media server.
- Configure the Net-Net SBC to enable Contact header mapping for the Refer-To header.

Caveats

The following caveats are associated with add-on conferencing:

- Contact header mapping is not replicated on the standby Net-Net SBC in an HA Net-Net SBC pair architecture.
- Upon switchover, any conferences in progress remain in progress, but no new parties can be invited to or join the conference.
- By default, the Net-Net SBC does not map SIP Contact headers for reasons of performance.

Add-On Conferencing Scenario

The add-on conferencing scenario described in the following example applies to a network architecture involving the Net-Net SBC and a media server that is located on a different network from the other conference participants. In this scenario, the Net-Net SBC resides on a standalone network that connects two additional, separate networks.

Some network architectures have a media server on a different network from the one on which the phones reside. In this scenario, all requests and/or responses going from the phones (Phone A, Phone B, or Phone C) to Media Server D and vice versa are translated according to their corresponding SIP NAT. All headers subjected to NAT are encoded and decoded properly as they traverse the Net-Net SBC, except for the Contact header. This exception occurs because the SIP process on the Net-Net SBC runs as a SIP B2BUA and not as a SIP proxy.

The SIP B2BUA re-originates the Contact headers of the User Agents (UAs) participating in SIP sessions with local Contact headers to make sure that they receive all future in-dialog requests. For an in-dialog request, the B2BUA can identify the dialog and find the Contact URI of the other leg of the call.

The Net-Net SBC add-on conferencing feature applies to situations when the Contact URI is used in another dialog. In such a case, the SIP B2BUA will not be able to find the correct dialog that retrieves the correct Contact URI of the other leg if it needs to replace the Contact URI.

Using the SIP add-on conferencing, the SIP B2BUA on the Net-Net SBC can map the Contact headers it receives to the Contact headers it creates. It can also convert the Refer-To URI to the correct value required for forwarding the REFER request.

SIP B2BUA Functionality

This section describes the role of the Net-Net SBC's SIP B2BUA in the add-on conferencing scenario that requires Contact header mapping for the Refer-To header.

When the Net-Net SBC starts up, the SIP B2BUA reads and parses the list of options in the SIP configuration. If the refer to uri prefix is an appropriate value (it is not an empty string), the Net-Net SBC will have a text prefix value the media server can use to denote a conference ID in its Contact header. With this information, the SIP B2BUA sets up a Contact header mapping.

You configure the Net-Net SBC to enable Contact header mapping for the Refer-To header by editing the SIP config options parameter. The SIP B2BUA on the Net-Net SBC can then map the Contact headers it receives to the Contact headers it creates.

Contact Header Processing

The Contact header mapping matches a Contact header that contains the refer to URI prefix to the corresponding Contact header that the Net-Net SBC's SIP B2BUA re-originates. Contact headers that do not contain the refer to URI prefix are not mapped (so that performance of the Net-Net SBC is minimally affected).

Only the Contact header in an INVITE request and its 200 OK response are checked for the refer to URI prefix and added to the Contact header mapping. Contact headers appearing in other SIP requests/responses are not checked.

Target Mapping and Conferences

If the Net-Net SBC is configured to enable Contact header mapping for the Refer-To header, then Contact header target maps are established for each individual call. The Net-Net SBC's SIP B2BUA uses these maps to allow the media server to connect the conference initiator with the conferenced-in parties.

Prior to terminating the call (hanging up), the conference initiator can contact other parties and invite those additional parties to join the conference. These other parties can join the existing conference because the target mapping for the conference is still in effect on the Net-Net SBC.

Once the conference initiator hangs up, the Net-Net SBC discards the mapping from the conference.

Refer-To Header Processing

When a Refer-To header is present in a REFER request that arrives at the SIP B2BUA after the incoming request is properly translated according to its SIP-NAT, the SIP B2BUA follows these steps:

1. The SIP B2BUA parses the Refer-To URI.
2. If the user part of the Refer-To URI contains the refer to URI prefix, the SIP B2BUA searches the Contact header mapping for a match of the user part of the URI.

If the user part of the Refer-To URI does not contain the refer to URI prefix, the SIP B2BUA leaves the existing Refer-To URI unchanged.

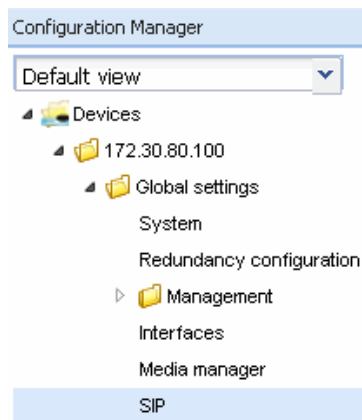
3. If the user part of the Refer-To URI contains the refer to URI prefix and a match of the Refer-To URI is found, the SIP B2BUA replaces the existing Refer-To URI with the URI of the corresponding Contact URI stored in the matched record. This replacement enables the NAT function to properly decode the replacement URI and change it back to the form originally received by the Net-Net SBC. As a result, the correct conference ID is restored in the Refer-To header prior to the request being sent to its next hop.

If the user part of the Refer-To URI contains the refer to URI prefix but a matched URI cannot be found, the SIP B2BUA will leave the existing Refer-To URI unchanged and will write a WARNING level log message to record the failure.

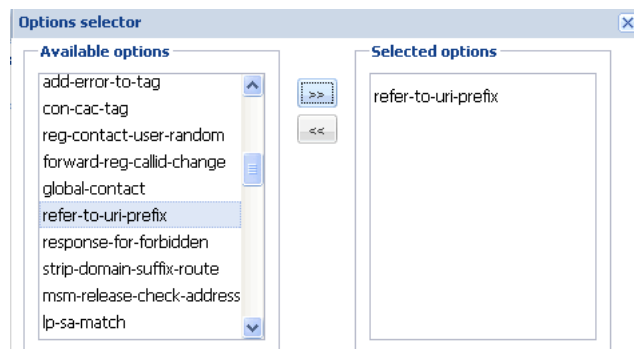
Configuring Add-On Conferencing

This section describes how to configure add-on conferencing on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The SIP options window appears.
5. Click **refer-to-uri-prefix** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.

7. **refer-to-uri-prefix**—Edit the Options textbox to append a prefix. For example:

Option

```
refer-to-uri-prefix="conf"
```

8. Click **OK** to save your edits and close the SIP options window.
9. Click **Apply**.

Roaming

This section explains how to configure SIP roaming. SIP roaming lets subscribers move from one active SIP device to another (at the same site or multiple sites) and retain service at the last registering device.

Overview

The Net-Net SBC supports multiple active registrations for the same user. The softswitch makes decisions regarding the current location of the user and the handling of requests from devices that are not currently identified as the user location. When there are multiple NATs, the Net-Net SBC is still required to let the softswitch be able to differentiate it.

The Net-Net SBC's SIP roaming ability supports the following features:

- Multiple active registrations from the same user can be cached, allowing subscribers to move from one active SIP device to another (at the same site or multiple sites) and still retain service at the last registering device. With the SIP roaming feature, one person, using multiple devices, can be contacted at all of the devices. These multiple devices (with their unique contact information) register to indicate that they are available for anyone that wants to contact that one person.
- The Net-Net SBC can also inform network devices (such as softswitches) of private SIP device IP addresses (endpoints) and the public firewall address of the user location.

Process Overview

Caller 1 wants to contact Person A. Caller 1 sends a message to `persona@acmepacket.com`, but Person A has configured more than one SIP-enabled device to accept messages sent to that address. These devices have unique addresses of `desk@10.0.0.4` and `phone2@10.0.0.5`. Person A has `desk@10.0.0.4` and `phone2@10.0.0.5` registered with the Net-Net SBC for anything addressed to `persona@acmepacket.com`.

With the SIP roaming feature, the Net-Net SBC accepts and stores both registrations for `persona@acmepacket.com`. That way, when someone wants to get in touch with Person A, the messages are sent to both devices (`desk@10.0.0.4` and `phone2@10.0.0.5`) until Person A answers one of them. You do not need to configure your Net-Net SBC for this functionality; your Net-Net SBC automatically provides it.

Using Private IP Addresses

In addition to supporting multiple registries, the Net-Net SBC can also distinguish user locations by their private IP address and the IP address of the public firewall. Using this information, the Net-Net SBC adds private endpoint and public firewall information to Contact headers.

For example, entering this information causes a Contact header that formerly appeared as the following:

`Contact: <sip:0274116202@63.67.143.217>`

to subsequently appear as the following:

`Contact: <sip:0274116202@63.67.143.217; ep=192.168.1.10; fw=10.1.10.21>`

The Net-Net SBC's SIP proxy reads this information and populates the `contact-endpoint` and `contact-firewall` fields with the appropriate values.

Example 1: With a NAT Firewall

The Net-Net SBC SIP proxy is configured with the following changeable parameters:

- endpoint= IP address of the SIP UA
- useradd= IP address of the Firewall Public IP address or the source layer 3 IP address of Register message
- userport= IP address port number of the Firewall Public IP address or the source layer 3 IP address port of Register message
- Net-Net SBC address=63.67.143.217
- firewall public address=10.1.10.21
- firewall public address port=10000
- SIP endpoint behind firewall=192.168.1.10

SIP message Contact header:

```
Contact:<sip:0274116202@63.67.143.217; endpoint=192.168.1.10;
useradd=10.1.10.21; userport=10000; transport=udp>
```

Example 2: Without a NAT Firewall

The Net-Net SBC SIP proxy is configured with the following changeable parameters:

- useradd= IP address of the SIP UA or the source layer 3 IP address of Register message
- userport= IP address port number of the SIP UA or the source layer 3 IP address port of Register message
- Net-Net SBC address=63.67.143.217
- SIP endpoint=192.168.1.10
- SIP endpoint IP address port=5060

SIP message Contact header:

```
Contact:<sip:0274116202@63.67.143.217; useradd=192.168.1.10;
userport=5060; transport=udp>
```

For SIP, the softswitch responsibility is that the URI SD put in the Contact of the REGISTER message should be reflected in the 200-OK response to the REGISTER request. The Contact header of the response should have an expires header parameter indicating the lifetime of the registration.

The following example shows a Net-Net SBC Send:

```
Contact: <sep: 0274116202@63.67.143.217 endpoint=192.168.1.10;
useradd=10.1.10.21; userport=10000>;
```

The following examples shows the softswitch Respond:

```
Contact: <sep: 0274116202@63.67.143.217 endpoint=192.168.1.10;
useradd=10.1.10.21; userport=10000>; expires=360
```

The contact field for endpoint and firewall parameters only appear in the following:

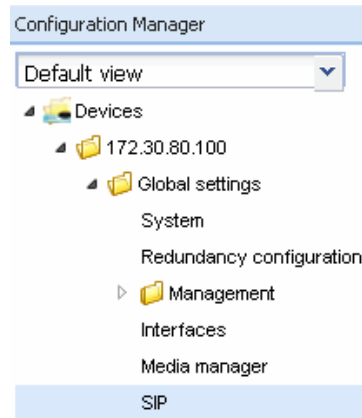
- Contact header of a REGISTER request sent from the Net-Net SBC to the softswitch server
- Contact header of a REGISTER response sent from the softswitch server to the Net-Net SBC
- Request-URI of an initial INVITE sent from the UT CSA server to the Net-Net SBC

An active endpoint is deleted when it does not register within the registration-interval setting or receives a 401 Unauthorized.

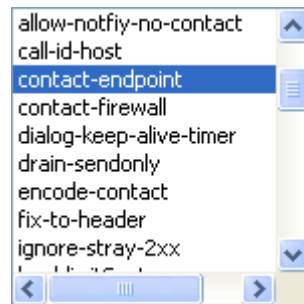
Configuring SIP Roaming

You can configure the SIP configuration's options parameter to indicate that you want to use the *private* IP address of the SIP device that the user is using and/or the *public* firewall address that identifies the location of the device. If defined, these options will be added as parameters to all Contact headers.

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The SIP options window appears.
5. Click **contact-endpoint** (or **contact-firewall**) in the Available options list to select it.



6. Click  to move the option to the Selected options list.
7. Repeat steps 5 and 6 if you want to select the second option (contact-endpoint or contact-firewall).
The option name(s) appear along with a description in the lower part of the window.
8. **contact-endpoint**—Edit the textbox to append a label for the information.

9. **contact-firewall**—Edit the textbox to append a label for the information. For example:

Option

```
contact-endpoint="endpoint"
```

Option

```
contact-firewall="firewall"
```

10. Click **OK** to save your edits and close the SIP options window.
11. Click **Apply**.

Embedded Header Support

This section explains how to configure embedded header support. The Net-Net SBC supports methods of extracting an embedded P-Asserted-Identity header from a contact header to support E911 when integrated with certain vendor's systems. See RFC 3455 - Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP) for more information.

The embedded header support feature watches for a specified embedded header contained in a Contact header received in a 3XX message. When the specified embedded header is found, the full <header=value> pair is inserted as a unique header in a redirected INVITE message that exits the Net-Net SBC. If the outgoing INVITE message were to contain the specified header, regardless of the use of this feature, the value extracted from the 3XX message replaces the INVITE message's specified header value.

If an incoming Contact header in a 3XX message looks like:

```
Contact: <ESRN@IPv4_Intrado_GW;user=phone?P-Asserted-Identity=%3Csip:+1-ESQK@IPv4_My_EAG;user=phone%3E>
```

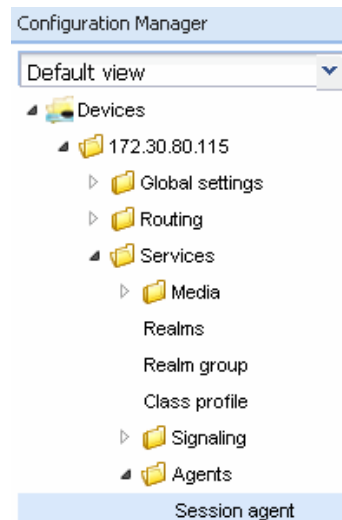
Then, if you configure your Net-Net SBC to parse for the embedded P-Asserted-Identity header to write as a unique header in the outgoing invite message, the outgoing INVITE and P-Asserted-Identity headers will look like:

```
INVITE SIP: ESRN@IPv4_Intrado_GW;user=phone
P-Asserted-Identity: +1-ESQK@IPv4_My_EAG;user=phone
```

Configuring Embedded Header Support

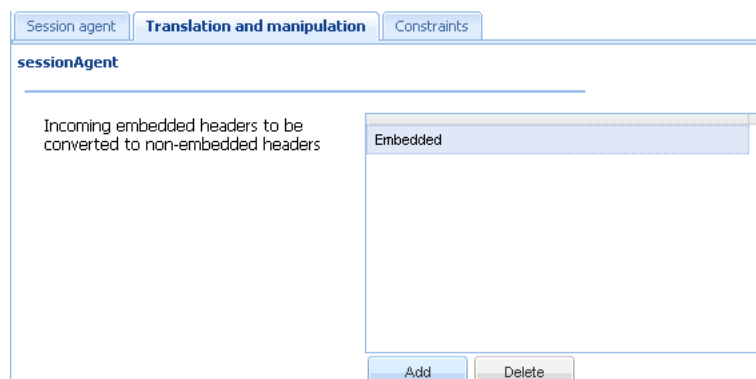
To configure embedded header support:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click a SIP session agent in the table to select it.
4. Click **Edit**.
5. Click the Translation and manipulation tab.

6. **Incoming embedded headers to be converted to non-embedded headers**—Click **Add** for Incoming embedded headers to be converted to non-embedded headers.



A dialog box appears.

7. Enter the name of the header you want to convert from embedded to non-embedded.
8. Click **OK**. The Add Embedded headers dialog box closes and the header name appears in the Embedded header list.
9. Click **Apply**.

Header and Parameter Manipulation

This section explains the SIP header and parameter manipulation feature, which lets the Net-Net SBC add, modify, and delete SIP headers and parts of SIP headers called SIP header elements. SIP header elements are the different subparts of the header, such as the header value, header parameter, URI parameter and so on (excluding the header name).

To enable the SIP header and parameter manipulation functionality, you create header manipulation rulesets in which you specify header manipulation rules, as well as optional header element rules that operate on specified header elements. You then apply the header manipulation ruleset as inbound or outbound for a session agent or SIP interface.

Header Manipulation Rules

Header manipulation rules operate on the header you specify when you configure the rule. A header manipulation rule can also be configured with a list of element rules, each of which would specify the actions you want performed for a given element of this header.

Header Element Rules

Header element rules perform operations on the elements of a header. Header elements include all subparts of a header; excluding the header name. For example, header value, header parameter, URI parameter, and so on.

About SIP Header and Parameter Manipulation

Using the SIP header manipulation ruleset, you can cause the Net-Net SBC to:

- Delete a header based on header name match.
- Delete a header based on header name match as well as header value match.

- Add a header.
- Modify the elements of a header (by configuring header element rules):
 - Add an element to a header.
For example, add a parameter to a header or add a URI parameter to the URI in a header.
 - Delete an element from a header.
For example, delete a parameter from a header or delete a URI parameter from the URI in a header.
 - Modify an element of a header.
For example, replace a FQDN with an IP address in a header or replace the value of a parameter in the header.
 - Delete a message body part
For example, delete the body part if the Content-Type is “application/ISUP”.

Role in Trunk Group URI Feature

SIP header and parameter manipulation plays a role in the trunk group URI feature. You need to set Route based on trunk group parameters to one of the trunk group values when configuring SIP header rules, if using this feature. (In addition you can configure session agents and session agents groups on the Net-Net SBC to insert trunk group URI parameters in the SIP contact header.

For all trunk group URI support, you must set the appropriate parameters in the SIP header manipulation configuration and in the session agent or session agent group configurations.

For trunk group URI support, the SIP header and parameter manipulation configuration tells the Net-Net SBC where and how to manipulate the SIP message to use originating (access) and terminating (egress) trunk group URI parameters.

See Trunk Group URIs for more information about trunk group URIs.

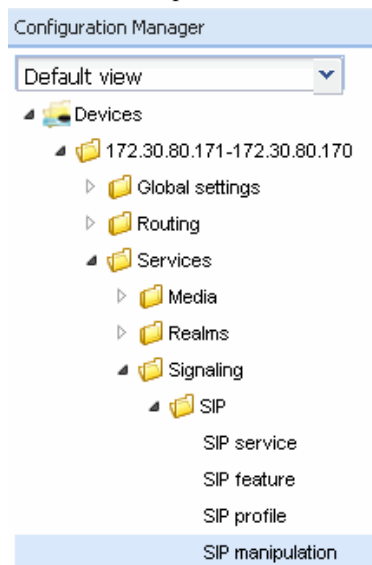
Configuring Header and Parameter Manipulation

This section explains how to configure SIP header and parameter manipulation. First you create a SIP header manipulation ruleset, then the header manipulation rules and optional header element rules you want that ruleset to contain. You then configure a session agent or a SIP interface to use the SIP header and parameter manipulation ruleset in the inbound and outbound directions.

Creating a SIP Header Manipulation Ruleset

To configure SIP header manipulation ruleset:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Click **Add**. The Add SIP header manipulation dialog box appears.
5. **SIP manipulation name**—Enter a name for the new SIP header manipulation ruleset. For example:

The screenshot shows the 'Add sipManipulation' dialog box. It has a title bar with the text 'Add sipManipulation'. Inside the dialog, there is a label '*SIP manipulation name' followed by a text input field containing the text 'sipheader1'. To the right of the input field is the text '(Unique required)'. At the bottom of the dialog, there are two buttons: 'Apply' and 'Cancel'.

6. Click **Apply** to save your new ruleset.
The new ruleset name appears at the top of the table.

Adding Header Rules

To add header rules

1. Click the new ruleset name in the table and click **Edit**.
2. Click **Add**. The Add Header rule dialog box appears.

- In the Service field, select headerRule from the combination box.
The Add Header, MIME and MIME ISUP rules dialog box appears.

Add Header, MIME and MIME ISUP rules

*Header rule name: sipheader1 (Unique required)

Sequence number: 1 (Range:-2147483648..2147483647. Default:-2147483648)

Header name: headertest

Apply Cancel

- Header rule name**—Enter the name of the header to which this rule applies. (The name must match the SIP header name to which this rule applies.)
- Header name**—Enter the name of this rule.
- Click **Apply**. The header rule appears in the table.
- Click the header rule name and click **Edit**.
- Action**—Select the action you want applied to the header from the combination box.
- Comparison type**—Select the comparison type that the match value uses.
- Match value**—Enter the value to be matched, if any.
- Message type**—Select the message type to which this header rule applies from the combination box.

Header name	headertest
Action	manipulate ▼
Comparison type	pattern-rule ▼
Match value	
Message type	any ▼

Adding Methods

To add methods:

- In the Methods area of the Edit SIP header manipulation dialog box, click **Add**. The Method dialog box appears.
- SIP method name(s)**—Enter the SIP method name(s) to which you want to apply this header rule. If entering multiple method names, separate them with commas. For example:
INVITE, ACK, BYE

This field is empty by default. If you leave the method field empty, the header-rule is applied to all methods.

3. Click **OK** to save the method. The name appears in the Methods list.
If you want to define optional element rules for your header rule, go to the next section. If not, continue to step 4.
4. Click **Apply** to save the header rule information. The header rule information appears in the Edit SIP header manipulation table.
5. Click **Apply** if you have finished defining your header rule.

Creating Element Rules

To create element rules:

1. In the Element rules area, click **Add**.
The Add Element rule dialog box appears.
2. **Element rule name**—Enter the name of the element to which this rule applies.
Note: The name you enter depends on the element type you enter in step 5. For uri-param, uri-user-param, and header-param it is the parameter name to be added, replaced, or deleted. For all other types, it serves to identify the element rule and any name can be used.
3. Click **OK**. The element rule appears in the table.
4. Click the element rule and click **Edit**.
5. **Type**—Click the type of element on which to perform the action in the combination box. The options are:

Option	Description
header-value	Entire value of the header.
header-param	Parameter portion of the header.
uri-user	User portion of the SIP URI.
uri-host	Host portion of the SIP URI.
uri-port	Port number portion of the SIP URI.
uri-param	Parameter included in the SIP URI.
uri-header	Header included in a request constructed from the URI.
uri-user-param	User parameter of the SIP URI.

6. **Action**—Click the action you want applied to the element if there is a match value in the combination box. The options are:

Option	Description
add	Add a new element, if it does not already exist.
replace	Replace the elements.
delete-element	Delete the specified element, if it exists. Based on the match value, if entered in step 6.
delete-header	Delete the specified header, if it exists.

7. **Comparison type**—Click the type of value that needs to be matched for the action to be performed in the combination box. The options are:

Option	Description
case-sensitive	Perform a case-sensitive comparison of the stored value against the header.
case-insensitive	Perform a case-insensitive comparison of the stored value against the header.
pattern-rule	Prior to version S-C6.2, evaluate a referenced value by name to determine if its regular expression matches TRUE or FALSE. For release S-C6.2, the Net-Net SBC performs a regular expression comparison. If you are upgrading, this means that existing configurations where pattern-rule was set will be changed to boolean if the Net-Net SBC can parse the match value in the rule.
boolean	Evaluate a referenced value by name to determine if its regular expression or Boolean expression matches TRUE or FALSE. Also set the boolean setting on a specific stored match to determine whether or not that match has a stored value.
refer-case-sensitive	For release S-C6.2, perform a case-sensitive comparison of the stored value (or pattern rule subgroup) against: a header parameter value, URI parameter value, or header value.
refer-case-insensitive	For release S-C6.2, perform a case-insensitive comparison of the stored value (or pattern rule subgroup) against: a header parameter value, URI parameter value, or header value.

8. **Match value**—Enter the value you want to match against the element value for an action to be performed.
9. **Match value type**—Click the type of value in the combination box that needs to be matched to the match-field entry for the action to be performed. The default value is ANY. The values are:

Option	Description
IP	Element value in the SIP message must be a valid IP address to be compared to the match-value field entry. If the match-value field is empty, any valid IP address is considered a match. If the element value is not a valid IP address, it is not considered a match.

Option	Description
FQDN	Element value in the SIP message must be a valid FQDN to be compared to the match-value field entry. If the match-value field is empty, any valid FQDN is considered a match. If the element value is not a valid FQDN, it is not considered a match.
ANY	Element value in the SIP message is compared with the match-value field entry. If the match-value field is empty, all values are considered a match.

10. **New value**—Enter the value for a new element or to replace a value for an existing element. You can enter an expression that includes a combination of absolute values, pre-defined parameters, and operators.

- Absolute values, with which you can use double quotes for clarity. You must escape all double quotes and back slashes that are part of an absolute value, and enclose the absolute value in double quotes.

For example:

`"sip:"+$TRUNK_GROUP+"."+$TRUNK_GROUP_CONTEXT`

- Pre-defined parameters always start with a \$. Valid pre-defined parameters are:

Parameter	Description
\$ORIGINAL	Original value of the element is used.
\$LOCAL_IP	IP address of the SIP interface on which the message was received for inbound manipulation; or sent on for outbound manipulation.
\$REMOTE_IP	IP address the message was received from for inbound manipulation; or being sent to for outbound manipulation.
\$REMOTE_VIA_HOST	Host from the top Via header of the message is used.
\$TRUNK_GROUP	Trunk group is used.
\$TRUNK_GROUP_CONTEXT	Trunk group context is used.

- Operators are:

Operator	Description
+	Append the value to the end. For example: "acme"+"packet" generates "acmepacket"
+^	Prepends the value. For example: "acme"+"^"packet" generates "packetacme"
-	Subtract at the end. For example: "112311"-"11" generates "1123"
.^	Subtract at the beginning. For example: "112311"-"^"11" generates "2311"

- Click **Apply** to save your edits.

sipManipulation->headerRule->elementRule

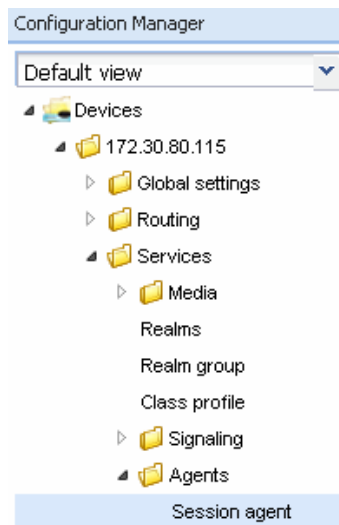
*Element rule name	acme
Parameter name	\$EMPTY
Type	uri-param
Action	add
Comparison type	case-sensitive
Match value	
Match value type	any
New value	

The new element rule appears in the Element rules table.

- Click **Apply** to return to the Edit SIP manipulation ruleset window.

Configuring a Session Agent

- Expand the Configuration Manager slider > Services > Agents.



- Click Session agent. The Session agent table appears in the content area.
- Select a session agent in the table and click **Edit**.
- Click the Translation and manipulation tab.
- Manipulation ruleset for inbound messages**—Select the SIP header manipulation ruleset you want to apply to SIP packets in inbound messages.
- Manipulation ruleset for outbound messages**—Select the SIP header manipulation ruleset you want to apply to SIP packets in outbound messages.

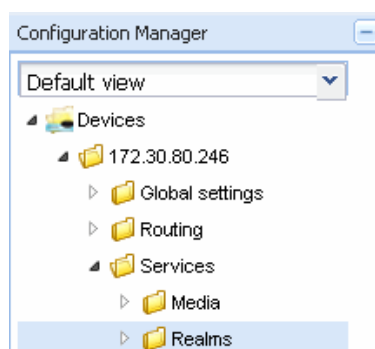
Manipulation ruleset for inbound messages	sipheader1
Manipulation ruleset for outbound messages	sipheader1

- Click **Apply**.

Configuring a SIP Interface

To configure a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm for which you want to configure the SIP interface in the table.
4. Click **Edit**.
5. Click the Registration tab and scroll to the Header manipulation section.
6. **Manipulation ruleset for inbound messages**—Select the SIP header manipulation ruleset you want to apply to SIP packets in inbound messages.
7. **Manipulation ruleset for outbound messages**—Select the SIP header manipulation ruleset you want to apply to SIP packets in outbound messages.

Manipulation ruleset for inbound messages ▼

Manipulation ruleset for outbound messages ▼

8. Click **Apply**.

Dialog-Matching Header Manipulation

The most common headers to manipulate using HMR are the To-URI and From-URI. Along with the to-tag, from-tag, and Call-ID values, these are also all headers that represent dialog-specific information that must match the UAC and UAS to be considered part of the same dialog. If these parameters are modified through HMR, the results can be that the UAC or UAS rejects messages.

While it is possible to ensure that dialog parameters match correctly using regular HMR, this feature offers a simpler and less error-prone method of doing so.

In addition, this section describes the addition of “built-in” SIP manipulations defined by Acme Packet best practices, and a new method of testing your SIP manipulations.

About Dialog-Matching Header Manipulations

The goal of this feature is to maintain proper dialog-matching through manipulation of dialog-specific information using HMR. Two fundamental challenges arise when looking at the issue of manipulating dialog-matching headers:

- Inbound HMR
- Outbound HMR

The **out-of-dialog** value for the **Message type** parameter addresses these challenges by offering an intelligent mode of dialog matching of messages for inbound and outbound HMR requests. This is a message type parameter, meaning that it becomes matching criteria for operations performed against a message. If you also specify methods (such as REGISTER) as matching criteria, then the rule is further limited to the designated method.

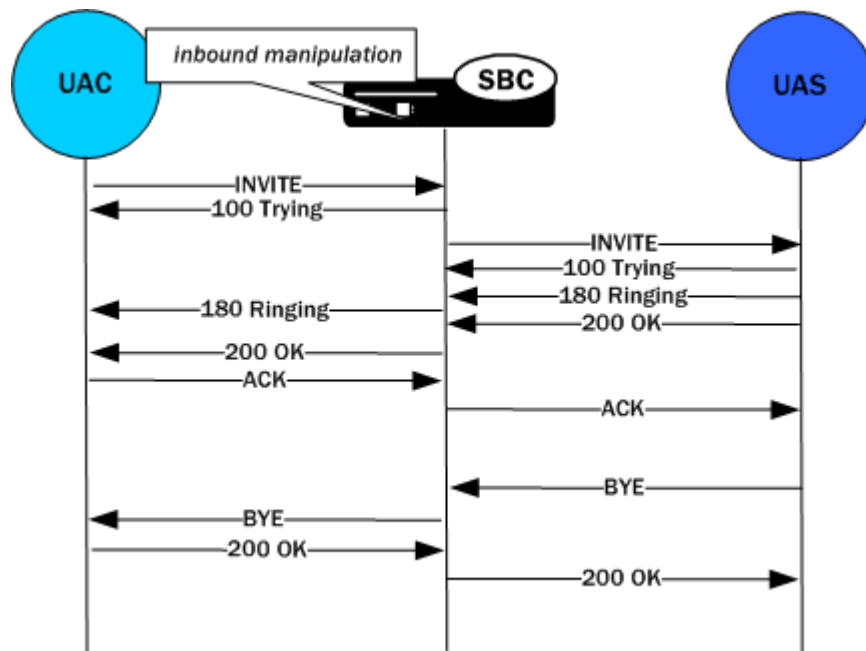
For both inbound and outbound manipulations, using the **out-of-dialog** value means the message must be a request without a to-tag in order to perform the manipulation.

Inbound HMR Challenge

Since inbound manipulations take place before the message reaches the core of Net-Net SBC SIP processing, the SIP proxy takes the manipulated header as what was directly received from the client. This can cause problems for requests leaving the Net-Net SBC for the UAC because the dialog will not match the initial request sent.

So the unmodified header must be cached because for any subsequent request (as in the case of a BYE originating from the terminator; see the diagram below) the Net-Net SBC might need to restore the original value—enabling the UAC to identify the message correctly as being part of the same dialog. For out-of-dialog requests (when the To, From, or Call-ID headers are modified) the original header will be stored in the dialog when **out-of-dialog** is used for the **Message type**.

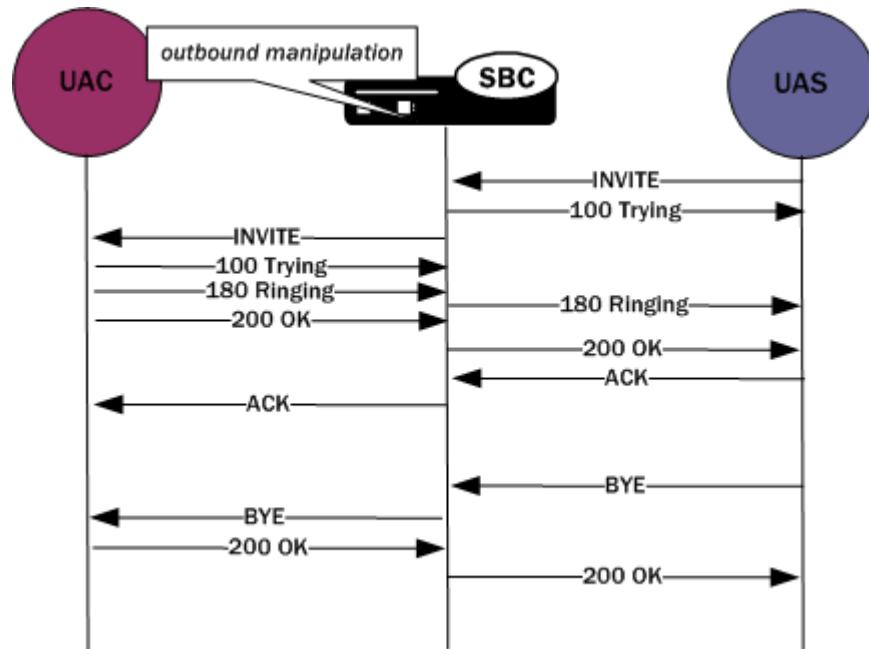
The Net-Net SBC performs the restoration of original headers outside of SIP manipulations. That is, there are no manipulation rules to configure for restore the header to their original context. The Net-Net SBC will recognize the headers have been modified, and restore them to their original state prior to sending the message out on the wire. Restoration takes place prior to outbound manipulations so that any outbound manipulation can those headers once they have been restored.



Outbound HMR Challenge

When you use the **out-of-dialog** value for an outbound manipulation, the Net-Net SBC only executes this specific SIP header rule only if the same SIP header rule was executed against the initial dialog-creating request.

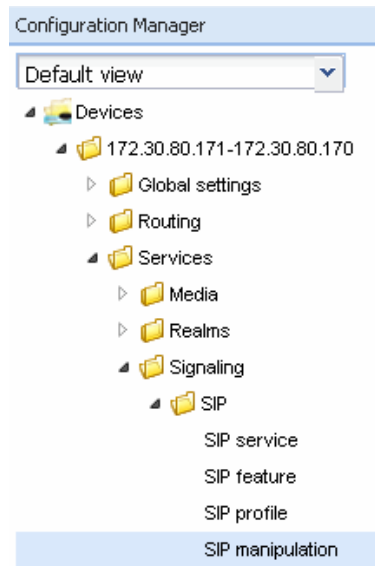
For example, if the INVITE's To header was not manipulated, it would not be correct to manipulate the To header in the BYE request. To do so would render the UAC unable to properly match the dialog. And this also means that the outbound manipulation should be carried out against a To, From, or Call-ID header in the INVITE request if it was manipulated in the INVITE.



Enabling Dialog-Matching Header Manipulation

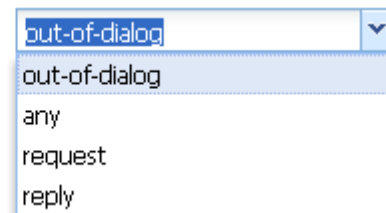
To enable dialog-matching header manipulation:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select a SIP manipulation name in the table and click **Edit**.
5. Select a header rule in the table and click **Edit**.
6. **Message type**—Click **out-of-dialog** in the combination box to enable dialog-matching header manipulation.

Message type



7. Click **Apply**. This brings you back to the Edit SIP header manipulation window.
8. Click **Apply** again.

Configuring Built-In SIP Manipulation

In the course of HMR use, certain rules have become commonly used. Lengthy and complex, these rules do not include any customer-specific information and they can be used widely. To make using them easier, they have been turned into built-in rules that you can reference in the **Manipulation ruleset for inbound messages** and **Manipulation ruleset for outbound messages** parameters that are part of the session agent, SIP interface, and realm configurations.

Built-in rules start with the prefix **ACME_**, so Acme Packet recommends you name your own rules in a different manner to avoid conflict.

While the number of built-in manipulation rules is expected to grow, one is supported at the present time: **ACME_NAT_TO_FROM_IP**. When performed outbound, this rule changes:

- To-URI hostname to the logical \$TARGET_IP and port to \$TARGET_PORT

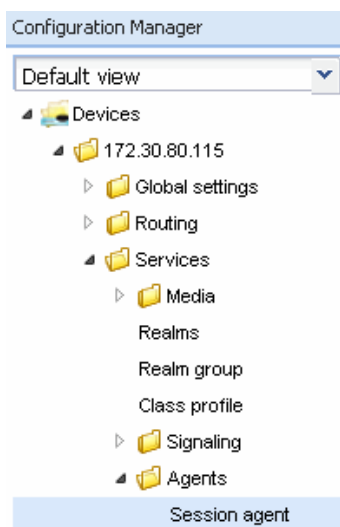
- From-URI to the logical \$REPLY_IP and port to be \$REPLY_PORT

The following sections explain how to enable this feature for a session agent, SIP interface, or realm by configuring the **Manipulation ruleset for inbound messages** or **Manipulation ruleset for outbound messages** parameters with the rule.

Session Agent

To configure built-in SIP manipulations for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent you want to configure and click **Edit**.
4. Click the Transaction and manipulation tab and scroll to Manipulation ruleset.
5. **Manipulation ruleset for inbound messages**—Click the name of the built-in rule you want to use for inbound messages in the combination box.
6. **Manipulation ruleset for outbound messages**—Click the name of the built-in rule you want to use for outbound messages in the combination box.

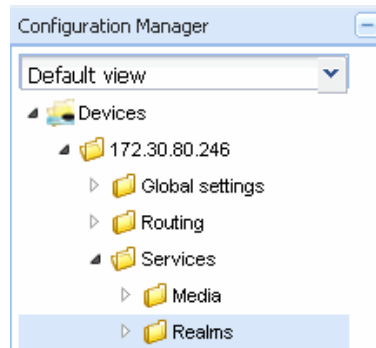
Manipulation ruleset for inbound messages	sipheader1
Manipulation ruleset for outbound messages	sipheader1

7. Click **Apply**.

SIP Interface

To configure built-in SIP manipulations for a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click the Registration tab at the top of the content area.
7. **Manipulation ruleset for inbound messages**—Click the name of the built-in rule you want to use for inbound messages in the combination box.
8. **Manipulation ruleset for outbound messages**—Click the name of the built-in rule you want to use for outbound messages in the combination box.

Header manipulation

Manipulation ruleset for inbound messages

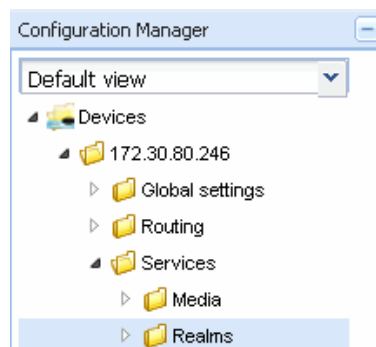
Manipulation ruleset for outbound messages

9. Click **Apply**.

Realm

To configure built-in SIP manipulations for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.

3. Select the SIP realm from the table to which you want to apply header manipulations and click **Edit**.
4. Click Realm tab and scroll to SIP manipulation.
5. **Manipulation ruleset for inbound messages**—Click the name of the built-in rule you want to use for inbound messages in the combination box.
6. **Manipulation ruleset for outbound messages**—Click the name of the built-in rule you want to use for outbound messages in the combination box.

Header manipulation

Manipulation ruleset for inbound messages

Manipulation ruleset for outbound messages

7. Click **Apply**.

Unique Header Manipulation Rule Regex Patterns

In addition to the Header Manipulation Rule (HMR) support it offers, the Net-Net SBC can now be provisioned with unique regex patterns for each logical remote entity. Unique Regex patterns saves you provisioning time and saves Net-Net SBC resources in instances when it was previously necessary to a unique SIP manipulation per PBX for a small number of customer-specific rules.

Manipulation Pattern Per Remote Entity

On the Net-Net SBC, you can configure logical remote entities (session agents, realms, and SIP interfaces) with a manipulation pattern string that the system uses as a regular expression. Then the SIP manipulation references this regular expression using the reserved word `$MANIP_PATTERN`. At runtime, the Net-Net SBC looks for the logical entity configured with a manipulation pattern string in this order of preference: session agent, realm, and finally SIP interface.

On finding the logical entity configured with the manipulation string, the Net-Net SBC dynamically determines the expression. This makes it possible for the same SIP manipulation to make a valid reference to a `$MANIP_PATTERN` for one call but not for the next. When there is an invalid reference to a manipulation pattern (either dynamically or by configuration error), the pattern-rule expression that results will turn out to be the default expression (which is `\, +`).

When the `$MANIP_PATTERN` is used in a manipulation rule's **new value** parameter, it resolves to an empty string, equivalent of no value. Even though this process ends with no value, it still consumes system resources. And so Acme Packet recommends you do not use `$MANIP_PATTERN` as a **new value** value.

Reject Action

When you use the new action type called **reject** to all manipulation rules and a condition matching the manipulation rule arises, the Net-Net SBC rejects the request (though it does not drop responses) and increments a counter. If the **Message type** parameter is set to **any** and the message is a:

- Response, the Net-Net SBC increments a counter to show the intention to reject the message—but the message will continue to be processed.

- Request, the Net-Net SBC performs the rejection and increments the counter.

The **New value** parameter supplies the status code and reason phrase corresponding to the reject. You can use the following syntax to supply this information: **status-code[:reason-phrase]**. You do not have to supply the status code and reason phrase information; by default, the system uses **400:Bad Request**.

If you do supply this information, then the status code must be a positive integer between 300 and 699. The Net-Net SBC then provides the reason phrase corresponding to the status code. And if there is no reason phrase, the system uses the one for the applicable reason class.

You can also customize a reason phrase. To do so, you enter the status code followed by a colon (:), being sure to enclose the entire entry in quotation marks. ("").

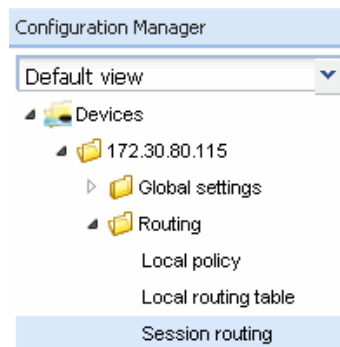
When the Net-Net SBC performs the **reject** action, the current SIP manipulation stops processing and does not act on any of the rules following the **reject** rule. This course of action is true for nested SIP manipulations that might have been constructed using the **sip-manip** action type.

Configuring the Reject Message Number and Time Window

To support the **reject** action, two parameters allow you to set how many messages in a certain amount of time cause the Net-Net SBC to generate an SNMP trap.

To configure the reject message number and time window:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. Scroll to the SIP manipulation counter section.
5. **Minimum reject message threshold to trigger trap**—Enter the minimum number of message rejections allowed in the **reject message window** time on the Net-Net SBC (when using the SIP manipulation action reject) before generating an SNMP trap. The default is 0, meaning this feature is disabled and no trap will be sent.
6. **Size of the reject message window (sec)**—Enter the time in seconds that defines the window for maximum message rejections allowed before generating

an SNMP trap. The default is 0, meaning this feature is disabled and no trap will be sent.

Minimum reject message threshold to trigger trap	0
Size of the reject message window (sec)	10

- Click **Apply**.

Log Action

When you use the action type called **log** and a condition matching the manipulation rule arises, the Net-Net SBC logs information about the current message to a separate log file. This log file will be located on the same core in which the SIP manipulation occurred, in the `sip` process. On the core where `sip` runs, a log file called `matched.log` will appear when this action type is executed.

The `matched.log` file contains a timestamp, received and sent Net-Net SBC network interface, sent or received IP address:port information, and the peer IP address:port information. It also specifies the rule that triggered the log action in this syntax: `rule-type[rule:name]`. The request URI, Contact header, To Header, and From header are also present.

```
-----
Apr 17 14:17:54.526 On [0:0]192.168.1.84:5060 sent to 192.168.1.60:5060
element-rule[checkRURIPort]
INVITE sip:service@192.168.1.84:5060 SIP/2.0
From: sipp <sip:+2125551212@192.168.1.60:5060>;tag=3035SIPpTag001
To: sut <sip:service@192.168.1.84>
Contact: sip:sipp@192.168.1.60:5060
```

Changes to Storing Pattern Rule Values

This release introduces changes to the framework for storing regular expression results within manipulation rules, altering the way the **store** action works. These changes are beneficial to performance.

In previous releases, when the **store** action is used, the Net-Net SBC stores all values matching the regular expression defined in the **match-value** parameter for all headers. At runtime, the system evaluates all stored values to find the correct index. And any could reference another rule's stored value.

Now, you no longer need to specify the **store** action. The simple fact of referencing another rule tells the system it must store a value. When SIP manipulation is used, the system first checks to see if any values require storing. The **add** action is an exception to this process; storing happens after a header is added.

Name Restrictions for Manipulation Rules

Historically, you have been allowed to configure any value in the following parameters:

- Header rules
- Element rules
- Mime rules
- Mime header rules
- Mime Isup rules

- ISUP param rules

This method of naming caused confusion when referencing rules, so now manipulation rules name must follow a specific syntax. They must match the expression `^[a-zA-Z0-9_]+$` and contain at least one lower case letter.

In other words, the name must:

- Start with a letter, and then it can contain any number of letters, numbers, or underscores
- Contain at least one lower case letter

All pre-existing configurations will continue to function normally. If you want to change a manipulation rule, however, you are required to change its name.

New Value Restrictions

To simplify configuration and remove possible ambiguity, the use of boolean and equality operators (`==`, `<=`, `<`, etc.) for **new value** parameter values has been banned. Since there was no specific functionality tied to their use, their ceasing to be use will have no impact to normal SIP manipulation operations.

Dialog Transparency

This section explains how to configure dialog transparency, which prevents the Net-Net SBC from generating a unique Call-ID and modifying dialog tags.

Overview

With dialog transparency enabled, the Net-Net SBC is prevented from generating a unique Call-ID and from modifying the dialog tags; the Net-Net SBC passes what it receives. Therefore, when a call made on one Net-Net SBC is transferred to another UA and crosses a second Net-Net SBC, the second Net-Net SBC does not note the context of the original dialog, and the original call identifiers are preserved end to end. The signalling presented to each endpoint remains in the appropriate context regardless of how many times a call crosses through a Net-Net SBC or how many Net-Net SBCs a call crosses.

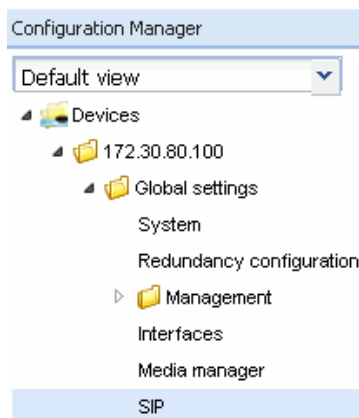
Without dialog transparency enabled, the Net-Net SBC's SIP B2BUA rewrites the Call-ID header and inserted dialog cookies into the From and To tags of all messages it processes. These dialog cookies are in the following format: `SDxxxxxNN-`. Using these cookies, the Net-Net SBC can recognize the direction of a dialog. However, this behavior makes call transfers problematic because one Net-Net SBC's Call-ID might not be properly decoded by another Net-Net SBC. The result is asymmetric header manipulation and failed call transfers.

Configuring Dialog Transparency

You set one parameter in your SIP configuration to enable dialog transparency.

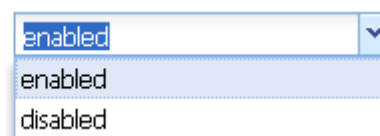
To enable SIP dialog transparency:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **Preserve CALL-IDs and tags**—Select enabled from the combination box to activate the dialog transparency feature. When enabled, the Net-Net SBC does not generate unique Call IDs and does not modify dialog tags.

Preserve Call-IDs and tags



5. Click **Apply**.

Route Header Removal

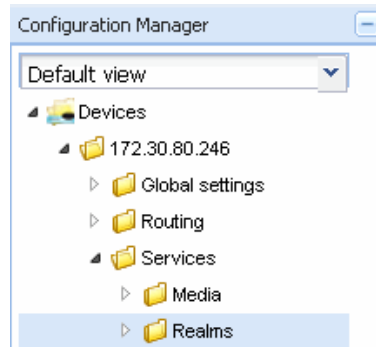
This section explains how to enable the Net-Net SBC to disregard and strip all SIP Route headers. You set an option in a SIP interface configuration to strip all Route headers for SIP requests coming from this interface.

When the Net-Net SBC with this option configured receives an INVITE from an interface, it removes the route headers. However, although it removes the headers, the Net-Net SBC maintains backward compatibility with RFC 2543 nodes. To do so, it normalizes the request to an RFC 3261 loose routing form before it removes the headers.

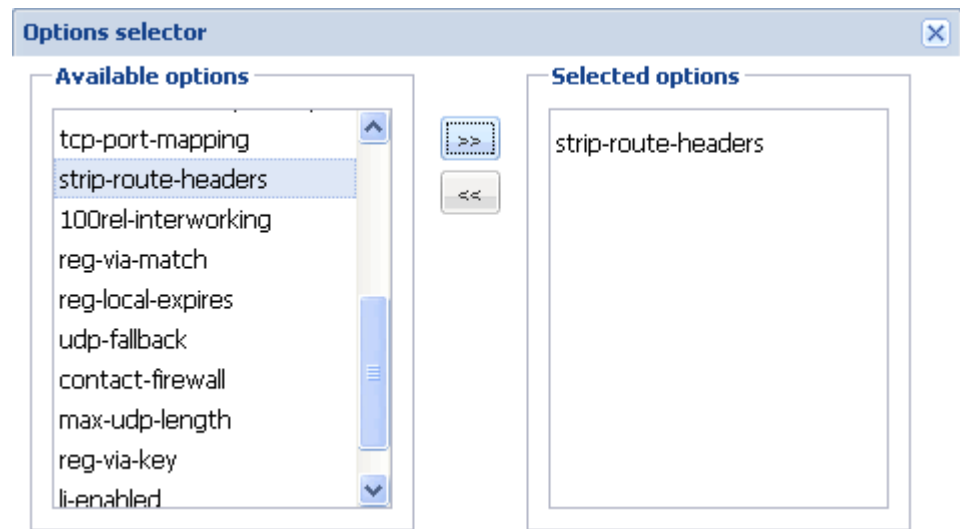
Configuring Route Header Removal

To configure SIP route header removal:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure and click **Edit**.
4. Click the Services and Protocols tab. The Services and Protocols table appears.
5. Click the name of the SIP interface to which you want to apply SIP route header removal in the table.
6. Click **Edit**.
7. Scroll to the Options in SIP service section and click **Options**. The Options dialog box appears.
8. Click **strip-route-headers** in the Available options list to select it.
9. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.



10. Click **OK**.
11. Click **Apply**.

Via Transparency

This section explains the inbound Via header transparency feature, which enables the Net-Net SBC to insert its Via header on top of the top-most Via header received from user equipment (UE). It then forwards it on to the IP Multimedia Subsystem (IMS) core with the original Via header now located as the bottom-most Via header.

The Net-Net SBC still replaces the Contact and other header addresses with its own, and does not pass on the core's Via headers in outbound requests.

This feature is targeted for the Telecoms & Internet converged Services & Protocols for Advanced Networks (TISPAN) with SIP hosted NAT traversal support. It works with SIP NAT bridged, local-policy routed, and non-SIP NAT configurations, regardless of registration handling.

How it Works

Some equipment acts as Proxy-CSCF (P-CSCF) and Serving-CSCF (S-CSCF) nodes, with the Net-Net SBC is located between the equipment and user endpoints. The equipment needs to see the each user endpoint's original Via header in order to perform some implicit authentication, admission, and control functions in a TISPAN-compliant model.

You enable Via header transparency on the access SIP interface or a session agent. Received Via headers are saved for inclusion in requests going out another interface or session agent that does not have the parameter set, in other words, the core side. For any received SIP message where the inbound previous hop interface was enabled for Via header transparency, the Net-Net SBC adds its own Via header as it forwards it, and it also copies the received top-most Via as the new bottom-most Via, if the outbound next hop interface/session agent is not enabled for Via header transparency. The Net-Net SBC also adds a received= parameter to the copied Via header, per the SIP RFC 3261.

Any message received from an interface without Via header transparency enabled, does not have the received Via header copied over to any other direction.

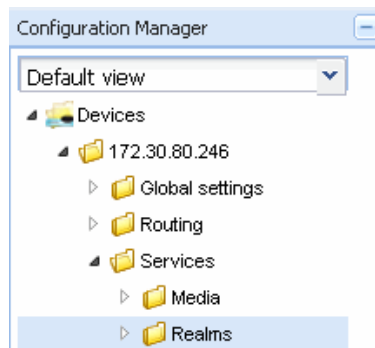
For HNT, where the original top-most (and only) Via header from a UE is a private/false address, the SD should still copy that false address into the core-side, and the received= parameter will contain the real Layer-3 addressing.

Configuring Via Transparency

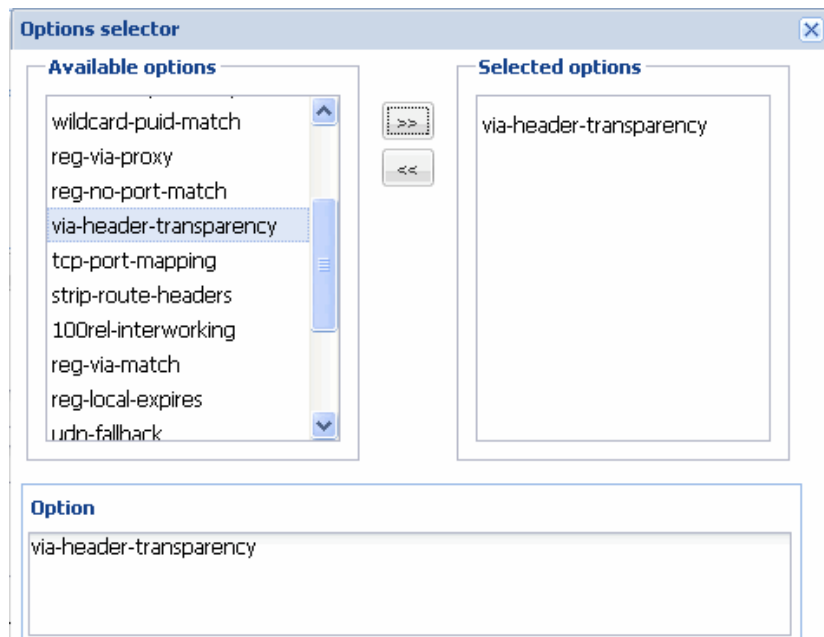
You can configure SIP Via header transparency for the access SIP interface using Net-Net Central.

To configure SIP Via header transparency for an access interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the table and click **Edit**.
4. Click the Service and Protocols tab. The Services and Protocols table appears.
5. Click the name of the realm to which you want to apply SIP via header transparency.
6. Click **Edit**.
7. Scroll to the Options in realm section and click **Options**. The SIP option window appears.
8. Click **via-header-transparency** in the Available options list to select it.
9. Click  to move the option to the opposite list. The option name appears along with a description in the lower part of the window.



10. Click **OK**.

11. Click **Apply**.

Symmetric Latching

Symmetric latching, or forced HNT, ensures that symmetric RTP/RTCP is used for a SIP endpoint. Symmetric RTP/RTCP means that the IP address and port pair used by an outbound RTP/RTCP flow is reused for the inbound flow. The IP address and port are learned when the initial RTP/RTCP flow is received by the Net-Net SBC. The flow's source address and port are latched onto and used as the destination for the RTP/RTCP sourced by the other side of the call. The IP address and port in the c line and m line respectively in the SDP message are ignored.

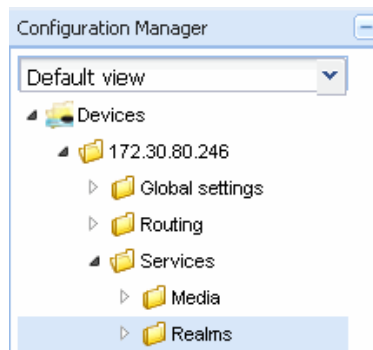
If your network is configured with nested realms in order to separate signalling from media, make sure that the symmetric latching feature is enabled on the signaling realm.

Note: This description is applicable to RTCP only when you also enable the HNT RTCP option in the **media-manager** configuration. Do not enable symmetric latching on core-facing interfaces.

Configuring Symmetric Latching

To configure symmetric latching:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Scroll to the Address translation profiles section.
4. **Symmetric latching**—Select enabled from the combination box.



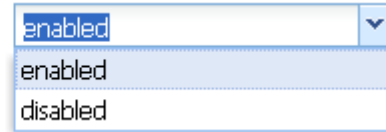
5. Click **Apply**.

Configuring RTCP Symmetric Latching

1. Expand the Global settings folder and click Media manager.
2. Click the CPU policing and DoS tab at the top of the content area.

3. **RTC support for NAT traversal**—Click the checkbox to enable HNT RTCP.

RTCP support for NAT traversal



4. Click **Apply**.

Number Normalization

This section explains the SIP number normalization feature that applies to the SIP To URI. (Currently the Net-Net SBC supports number normalization on From and To addresses for both inbound and outbound call legs.) Number normalization includes add, delete, and replace string functions that result in consistent number formats.

Number normalization is supported for the following call types:

- SIP to SIP
- H.323 to SIP

How it Works

Number normalization applies to the SIP To URI. It occurs on ingress traffic, prior to the generation of accounting records or local policy lookups. RADIUS CDR attributes are populated with the normalized numbers. Local policy matching is based on the normalized numbers.

Terminology

The following terminology is used in the descriptions contained in the next sections.

- X is any digit having the value 0 through 9
- N is any digit having the value 2 through 9
- 0/1 is a digit having the value of either 0 or 1
- NXX is a form of Numbering Plan Area (NPA).
- CC is a 1, 2, or 3 digit country code used in international dialing
- NN is a national number that can be a four to fourteen digit national number used in international dialing, where the combination of CC+NN is a 7 to 15 digit number.
- + symbol in E.164 indicates that an international prefix is required
- E.164 numbers are globally unique, language independent identifiers for resources on Public Telecommunication Networks that can support many different services and protocols.
- N11 number is any of the three-digit dialing codes in the form N11 used to connect users to special services, where N is a digit between 2 and 9

Calls from IP Endpoints

The Net-Net SBC uses the following number normalization rules:

- North American Numbering Plan (NANP) calls: where a number with the format **1NPANXXXXXX** is received, the Net-Net SBC adds a plus sign (+) as a prefix to the NANP number. The Net-Net SBC also adds the string **;user=phone** after the host IP address in the SIP URI. For example:

`sip:+1NPANXXXXXX@ipaddr;user=phone`

- International NWZ1 calls: Net-Net SBC receives an international call with the format `011CCNN`. The Net-Net SBC deletes the `011` prefix and adds a plus sign (+) as a prefix to `CC+NN`; and also adds the string `;user=phone` after the host IP address in the SIP URI. For example:

`sip:+CCNN@ipaddr;user=phone`

- Private number calls: when a private number with the format `nxxxx` (where `n=2` through `9`) is received, no number normalization is applied by the Net-Net SBC.
- Calls to numbers such as `N11`, `0-`, `0+`, `00-`, and `01+`: the Net-Net SBC adds `;phone-context=+1` after the number and also adds the string `;user=phone` after the host IP address in the SIP URI. For example:

`sip:N11;phone-context=+1@ipaddr;user=phone`

`sip:01CCNN;phone-context=+1@ipaddr;user=phone`

- Calls with numbers that are already normalized are not modified by the Net-Net SBC.

Calls from IP Peer Network

For calls received from external peer networks, the Net-Net SBC uses the following number normalization rules:

- Global numbers such as NANP and international E.164 numbers should have already been normalized. If not, the Net-Net SBC applies the same number normalization rules listed in the prior section.
- Calls to numbers such as `N11`, `0-`, `0+`, `00-`, and `01+`: the Net-Net SBC adds `;phone-context=+1` after the number and also adds the string `;user=phone` (if absent) after the host IP address in the SIP URI.

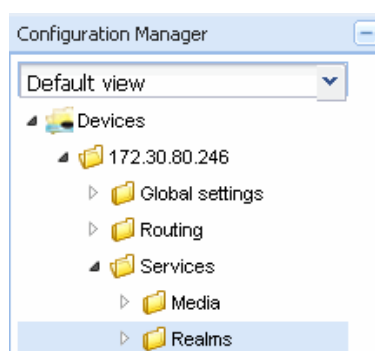
Configuring Number Normalization

You can configure SIP number normalization for the realm and session agent using Net-Net Central.

Realm

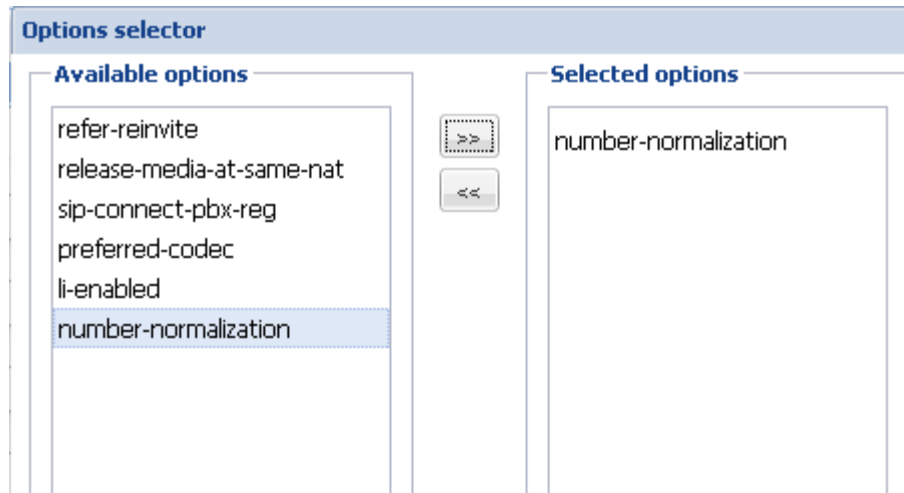
To configure SIP number normalization for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm and click **Edit**.
4. Scroll to the Options section and click **Options**. The Options dialog box appears.

5. Click **number-normalization** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.

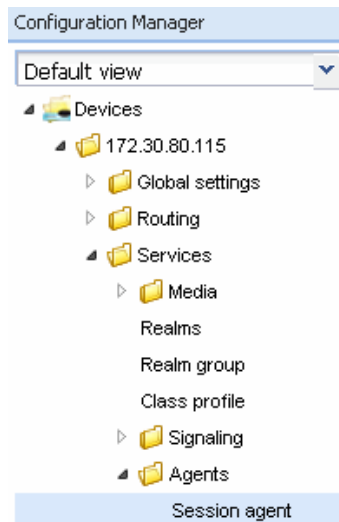


7. Click OK.
8. Click Apply.

Configuring Number Normalization for a Session Agent

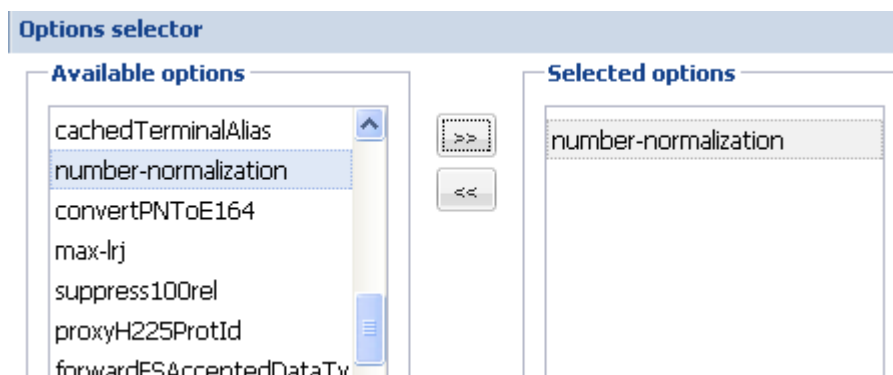
To configure SIP number normalization for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the SIP session agent to which you want to apply number normalization in the table and click **Edit**.
4. Scroll to the SIP message handling section.
5. In the SIP options for this SA parameter, click **Options**. The Options dialog box appears.

- Click **number-normalization** in the Select options list.



- Click **>>** to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
- Click **OK** to continue. The SIP options window closes and the option name appears in the SIP options for this SA list.
- Click **Apply**.

Ensuring Placement of RADIUS Attributes in CSV File

Using the Net-Net SBC's local CDR storage feature, you can enable the Net-Net SBC to write CDRs as comma-delimited ASCII records to files on the Net-Net SBC. The types of records are controlled by the same accounting configuration parameters used for RADIUS; the fields of the comma-delimited entries correspond to RADIUS Start, Interim, and Stop records. This feature is mainly used with the Net-Net SBC's FTP push capability, so that the files stored locally are periodically sent to an FTP server you designate.

By default, unpopulated or unused fields in the RADIUS CDR are omitted from the locally-stored CSV file; there is no fixed position for a RADIUS attribute across all CSV files. Instead, the missing values are skipped in the CSV file causing the order and appearance for attribute values to differ from record to record.

You can guarantee the placement of attributes in locally-stored CSV files by enabling the **Include empty fields in each CDR generated** parameter. With this parameter enabled, RADIUS records sent to a RADIUS client contain even empty attributes with an integer, date and time, or IP address format; the default value is zero. When there is no value to report:

- IP address attribute reports as 0.0.0.0
- Date and time attribute reports as 00:00:00.000 UTC JAN 01 1970
- Integer attribute value reports as 0

To maintain RFC 2865 and 2866 compliance, the Net-Net SBC will not send empty attributes that are string values to a RADIUS client. And when you enable this feature, the Net-Net SBC adds all attributes to the locally-stored CSV file.

Requirements

If you want to use this feature and guarantee the CSV placement for RADIUS attribute values, you must use the entire RADIUS dictionary. You cannot use the RADIUS CDR abbreviation feature. Using an abbreviated form of the RADIUS dictionary results in adverse effects for the CSV file.

You must set the **Included VSA ID range** parameter to use the entire range of attributes. Leaving this parameter blank disables abbreviation and all attributes are included. Alternatively, you can specify all of the parameters (by attribute number) that are used in the Net-Net OS release loaded on your system.

Recommended Configuration

Acme Packet recommends that you do not leave the **Included VSA ID range** parameter empty. You should set it to the attributes used in the Net-Net OS release loaded on your system. Your entry would look similar to the following:

1-4,10-14,20-24,28,29,32-71,74-136

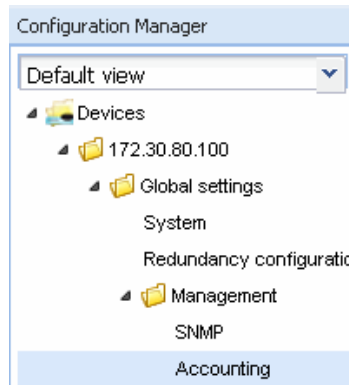
This entry is the same one shown in the configuration sample below.

Configuring Fixed Value Placement in CSV Files

This section shows you how to configure the CDR file enhancement to use fixed value placement in locally-stored CSV files.

To enable fixed value placement in CSV files for RADIUS CDRs:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Click the Radius accounting tab at the top of the content area.
5. **Included VSA ID range**—Either leave this parameter blank or enter the complete range of VSAs for the Net-Net OS release loaded on your system. The following example shows what you would enter to use all of the VSAs for Net-Net OS Release 4.1.4p4 for a system that is not running QoS.

Included VSA ID range

6. Click **Apply**.

CSV File Placement for RADIUS Attribute Values

This section contains three tables that show where in a locally-generated CSV file specific attributes appear—one each for Start, Interim, and Stop records.

Start Record CSV Placement

CSV Placement	Attribute Name	VSA ID Number
1	Acct-Status-Type	
2	NAS-IP-Address	
3	NAS-Port	
4	Acct-Session-Id	
5	Acme-Session-Ingress-CallId	3
6	Acme-Session-Egress-CallId	4
7	Acme-Session-Protocol-Type	43
8	Acme-Session-Generic-Id	40
9	Calling-Station-Id	
10	Called-Station-Id	
11	h323-setup-time	
12	h323-connect-time	
13	Acme-Session-Egress-Realm	42
14	Acme-Session-Ingress-Realm	41
15	Acme-FlowId_FS1_F	1
16	Acme-FlowType_FS1_F	2
17	Acme-Flow-In-Realm_FS1_F	10
18	Acme-Flow-In-Src-Addr_FS1_F	11
19	Acme-Flow-In-Src-Port_FS1_F	12
20	Acme-Flow-In-Dst-Addr_FS1_F	13
21	Acme-Flow-In-Dst-Port_FS1_F	14
22	Acme-Flow-Out-Realm_FS1_F	20
23	Acme-Flow-Out-Src-Addr_FS1_F	21
24	Acme-Flow-Out-Src-Port_FS1_F	22
25	Acme-Flow-Out-Dst-Addr_FS1_F	23
26	Acme-Flow-Out-Dst-Port_FS1_F	24
27	Acme-FlowID_FS1_R	78
28	Acme-FlowType_FS1_R	79
29	Acme-Flow-In-Realm_FS1_R	80
30	Acme-Flow-In-Src-Addr_FS1_R	81

CSV Placement	Attribute Name	VSA ID Number
31	Acme-Flow-In-Src-Port_FS1_R	82
32	Acme-Flow-In-Dst-Addr_FS1_R	83
33	Acme-Flow-In-Dst-Port_FS1_R	84
34	Acme-Flow-Out-Realm_FS1_R	85
35	Acme-Flow-Out-Src-Addr_FS1_R	86
36	Acme-Flow-Out-Src-Port_FS1_R	87
37	Acme-Flow-Out-Dst-Addr_FS1_R	88
38	Acme-Flow-Out-Dst-Port_FS1_R	89
39	Acme-FlowID_FS2_F	90
40	Acme-FlowType_FS2_F	91
41	Acme-Flow-In-Realm_FS2_F	92
42	Acme-Flow-In-Src-Addr_FS2_F	93
43	Acme-Flow-In-Src-Port_FS2_F	94
44	Acme-Flow-In-Dst-Addr_FS2_F	95
45	Acme-Flow-In-Dst-Port_FS2_F	96
46	Acme-Flow-Out-Realm_FS2_F	97
47	Acme-Flow-Out-Src-Addr_FS2_F	98
48	Acme-Flow-Out-Src-Port_FS2_F	99
49	Acme-Flow-Out-Dst-Addr_FS2_F	100
50	Acme-Flow-Out-Dst-Port_FS2_F	101
51	Acme-FlowID_FS2_R	112
52	Acme-FlowType_FS2_R	113
53	Acme-Flow-In-Realm_FS2_R	114
54	Acme-Flow-In-Src-Addr_FS2_R	115
55	Acme-Flow-In-Src-Port_FS2_R	116
56	Acme-Flow-In-Dst-Addr_FS2_R	117
57	Acme-Flow-In-Dst-Port_FS2_R	118
58	Acme-Flow-Out-Realm_FS2_R	119
59	Acme-Flow-Out-Src-Addr_FS2_R	120
60	Acme-Flow-Out-Src-Port_FS2_R	121
61	Acme-Flow-Out-Dst-Addr_FS2_R	122
62	Acme-Flow-Out-Dst-Port_FS2_R	123
63	Acme-Session-Charging-Vector	54

CSV Placement	Attribute Name	VSA ID Number
64	Acme-Session-Charging-Function_Address	55
65	Acme-Firmware-Version	56
66	Acme-Local-Time-Zone	57
67	Acme-Post-Dial-Delay	58
68	Acme-Primary-Routing-Number	64
69	Acme-Originating-Trunk-Group	65
70	Acme-Terminating-Trunk-Group	66
71	Acme-Originating-Trunk-Context	67
72	Acme-Terminating-Trunk-Context	68
73	Acme-P-Asserted-ID	69
74	Acme-Ingress-Local-Addr	74
75	Acme-Ingress-Remote-Addr	75
76	Acme-Egress-Local-Addr	76
77	Acme-Egress-Remote-Addr	77
78	Acme-SIP-Diversion	70
79	Acme-Egress-Final-Routing-Number	134
80	Acme-Session-Ingress-RPH	135
81	Acme-Session-Egress-RPH	136

Interim Record CSV Placement

CSV Placement	Attribute Name	VSA ID Number
1	Acct-Status-Type	
2	NAS-IP-Address	
3	NAS-Port	
4	Acct-Session-Id	
5	Acme-Session-Ingress-CallId	3
6	Acme-Session-Egress-CallId	4
7	Acme-Session-Protocol-Type	43
8	Acme-Session-Generic-Id	40
9	Calling-Station-Id	
10	Called-Station-Id	
11	h323-setup-time	

CSV Placement	Attribute Name	VSA ID Number
12	h323-connect-time	
13	Acme-Session-Egress-Realm	42
14	Acme-Session-Ingress-Realm	41
15	Acme-FlowId_FS1_F	1
16	Acme-FlowType_FS1_F	2
17	Acme-Flow-In-Realm_FS1_F	10
18	Acme-Flow-In-Src-Addr_FS1_F	11
19	Acme-Flow-In-Src-Port_FS1_F	12
20	Acme-Flow-In-Dst-Addr_FS1_F	13
21	Acme-Flow-In-Dst-Port_FS1_F	14
22	Acme-Flow-Out-Realm_FS1_F	20
23	Acme-Flow-Out-Src-Addr_FS1_F	21
24	Acme-Flow-Out-Src-Port_FS1_F	22
25	Acme-Flow-Out-Dst-Addr_FS1_F	23
26	Acme-Flow-Out-Dst-Port_FS1_F	24
27	Acme-Calling-RTCP-Packets-Lost_FS1	32
28	Acme-Calling-RTCP-Avg-Jitter_FS1	33
29	Acme-Calling-RTCP-Avg-Latency_FS1	34
30	Acme-Calling-RTCP-MaxJitter_FS1	35
31	Acme-Calling-RTCP-MaxLatency_FS1	36
32	Acme-Calling-RTP-Packets-Lost_FS1	37
33	Acme-Calling-RTP-Avg-Jitter_FS1	38
34	Acme-Calling-RTP-MaxJitter_FS1	39
35	Acme-Calling-Octets_FS1	28
36	Acme-Calling-Packets_FS1	29
37	Acme-FlowID_FS1_R	78
38	Acme-FlowType_FS1_R	79
39	Acme-Flow-In-Realm_FS1_R	80
40	Acme-Flow-In-Src-Addr_FS1_R	81
41	Acme-Flow-In-Src-Port_FS1_R	82
42	Acme-Flow-In-Dst-Addr_FS1_R	83
43	Acme-Flow-In-Dst-Port_FS1_R	84
44	Acme-Flow-Out-Realm_FS1_R	85

CSV Placement	Attribute Name	VSA ID Number
45	Acme-Flow-Out-Src-Addr_FS1_R	86
46	Acme-Flow-Out-Src-Port_FS1_R	87
47	Acme-Flow-Out-Dst-Addr_FS1_R	88
48	Acme-Flow-Out-Dst-Port_FS1_R	89
49	Acme-Called-RTCP-Packets-Lost_FS1	46
50	Acme-Called-RTCP-Avg-Jitter_FS1	47
51	Acme-Called-RTCP-Avg-Latency_FS1	48
52	Acme-Called-RTCP-MaxJitter_FS1	49
53	Acme-Called-RTCP-MaxLatency_FS1	50
54	Acme-Called-RTP-Packets-Lost_FS1	51
55	Acme-Called-RTP-Avg-Jitter_FS1	52
56	Acme-Called-RTP-MaxJitter_FS1	53
57	Acme-Called-Octets_FS1	44
58	Acme-Called-Packets_FS1	45
59	Acme-FlowID_FS2_F	90
60	Acme-FlowType_FS2_F	91
61	Acme-Flow-In-Realm_FS2_F	92
62	Acme-Flow-In-Src-Addr_FS2_F	93
63	Acme-Flow-In-Src-Port_FS2_F	94
64	Acme-Flow-In-Dst-Addr_FS2_F	95
65	Acme-Flow-In-Dst-Port_FS2_F	96
66	Acme-Flow-Out-Realm_FS2_F	97
67	Acme-Flow-Out-Src-Addr_FS2_F	98
68	Acme-Flow-Out-Src-Port_FS2_F	99
69	Acme-Flow-Out-Dst-Addr_FS2_F	100
70	Acme-Flow-Out-Dst-Port_FS2_F	101
71	Acme-Calling-RTCP-Packets-Lost_FS2	104
72	Acme-Calling-RTCP-Avg-Jitter_FS2	105
73	Acme-Calling-RTCP-Avg-Latency_FS2	106
74	Acme-Calling-RTCP-MaxJitter_FS2	107
75	Acme-Calling-RTCP-MaxLatency_FS2	108
76	Acme-Calling-RTP-Packets-Lost_FS2	109
77	Acme-Calling-RTP-Avg-Jitter_FS2	110

CSV Placement	Attribute Name	VSA ID Number
78	Acme-Calling-RTP-MaxJitter_FS2	111
79	Acme-Calling-Octets_FS2	102
80	Acme-Calling-Packets_FS2	103
81	Acme-FlowID_FS2_R	112
82	Acme-FlowType_FS2_R	113
83	Acme-Flow-In-REALM_FS2_R	114
84	Acme-Flow-In-Src-Addr_FS2_R	115
85	Acme-Flow-In-Src-Port_FS2_R	116
86	Acme-Flow-In-Dst-Addr_FS2_R	117
87	Acme-Flow-In-Dst-Port_FS2_R	118
88	Acme-Flow-Out-REALM_FS2_R	119
89	Acme-Flow-Out-Src-Addr_FS2_R	120
90	Acme-Flow-Out-Src-Port_FS2_R	121
91	Acme-Flow-Out-Dst-Addr_FS2_R	122
92	Acme-Flow-Out-Dst-Port_FS2_R	123
93	Acme-Called-RTCP-Packets-Lost_FS2	126
94	Acme-Called-RTCP-Avg-Jitter_FS2	127
95	Acme-Called-RTCP-Avg-Latency_FS2	128
96	Acme-Called-RTCP-MaxJitter_FS2	129
97	Acme-Called-RTCP-MaxLatency_FS2	130
98	Acme-Called-RTP-Packets-Lost_FS2	131
99	Acme-Called-RTP-Avg-Jitter_FS2	132
100	Acme-Called-RTP-MaxJitter_FS2	133
101	Acme-Called-Octets_FS2	124
102	Acme-Called-Packets_FS2	125
103	Acme-Session-Charging-Vector	54
104	Acme-Session-Charging-Function_Address	55
105	Acme-Firmware-Version	56
106	Acme-Local-Time-Zone	57
107	Acme-Post-Dial-Delay	58
108	Acme-Primary-Routing-Number	64
109	Acme-Originating-Trunk-Group	65
110	Acme-Terminating-Trunk-Group	66

CSV Placement	Attribute Name	VSA ID Number
111	Acme-Originating-Trunk-Context	67
112	Acme-Terminating-Trunk-Context	68
113	Acme-P-Asserted-ID	69
114	Acme-Ingress-Local-Addr	74
115	Acme-Ingress-Remote-Addr	75
116	Acme-Egress-Local-Addr	76
117	Acme-Egress-Remote-Addr	77
118	Acme-SIP-Diversion	70
119	Acme-Intermediate_Time	63
120	Acme-Egress-Final-Routing-Number	134
121	Acme-Session-Ingress-RPH	135
122	Acme-Session-Egress-RPH	136

Stop Record CSV Placement

CSV Placement	Attribute Name	VSA ID Number
1	Acct-Status-Type	
2	NAS-IP-Address	
3	NAS-Port	
4	Acct-Session-Id	
5	Acme-Session-Ingress-CallId	3
6	Acme-Session-Egress-CallId	4
7	Acme-Session-Protocol-Type	43
8	Acme-Session-Generic-Id	40
9	Calling-Station-Id	
10	Called-Station-Id	
11	Acct-Terminate-Cause	
12	Acct-Session-Time	
13	h323-setup-time	
14	h323-connect-time	
15	h323-disconnect-time	
16	h323-disconnect-cause	
17	Acme-Session-Egress-Realm	42

CSV Placement	Attribute Name	VSA ID Number
18	Acme-Session-Ingress-Realm	41
19	Acme-FlowId_FS1_F	1
20	Acme-FlowType_FS1_F	2
21	Acme-Flow-In-Realm_FS1_F	10
22	Acme-Flow-In-Src-Addr_FS1_F	11
23	Acme-Flow-In-Src-Port_FS1_F	12
24	Acme-Flow-In-Dst-Addr_FS1_F	13
25	Acme-Flow-In-Dst-Port_FS1_F	14
26	Acme-Flow-Out-Realm_FS1_F	20
27	Acme-Flow-Out-Src-Addr_FS1_F	21
28	Acme-Flow-Out-Src-Port_FS1_F	22
29	Acme-Flow-Out-Dst-Addr_FS1_F	23
30	Acme-Flow-Out-Dst-Port_FS1_F	24
31	Acme-Calling-RTCP-Packets-Lost_FS1	32
32	Acme-Calling-RTCP-Avg-Jitter_FS1	33
33	Acme-Calling-RTCP-Avg-Latency_FS1	34
34	Acme-Calling-RTCP-MaxJitter_FS1	35
35	Acme-Calling-RTCP-MaxLatency_FS1	36
36	Acme-Calling-RTP-Packets-Lost_FS1	37
37	Acme-Calling-RTP-Avg-Jitter_FS1	38
38	Acme-Calling-RTP-MaxJitter_FS1	39
39	Acme-Calling-Octets_FS1	28
40	Acme-Calling-Packets_FS1	29
41	Acme-FlowID_FS1_R	78
42	Acme-FlowType_FS1_R	79
43	Acme-Flow-In-Realm_FS1_R	80
44	Acme-Flow-In-Src-Addr_FS1_R	81
45	Acme-Flow-In-Src-Port_FS1_R	82
46	Acme-Flow-In-Dst-Addr_FS1_R	83
47	Acme-Flow-In-Dst-Port_FS1_R	84
48	Acme-Flow-Out-Realm_FS1_R	85
49	Acme-Flow-Out-Src-Addr_FS1_R	86
50	Acme-Flow-Out-Src-Port_FS1_R	87

CSV Placement	Attribute Name	VSA ID Number
51	Acme-Flow-Out-Dst-Addr_FS1_R	88
52	Acme-Flow-Out-Dst-Port_FS1_R	89
53	Acme-Called-RTCP-Packets-Lost_FS1	46
54	Acme-Called-RTCP-Avg-Jitter_FS1	47
55	Acme-Called-RTCP-Avg-Latency_FS1	48
56	Acme-Called-RTCP-MaxJitter_FS1	49
57	Acme-Called-RTCP-MaxLatency_FS1	50
58	Acme-Called-RTP-Packets-Lost_FS1	51
59	Acme-Called-RTP-Avg-Jitter_FS1	52
60	Acme-Called-RTP-MaxJitter_FS1	53
61	Acme-Called-Octets_FS1	44
62	Acme-Called-Packets_FS1	45
63	Acme-FlowID_FS2_F	90
64	Acme-FlowType_FS2_F	91
65	Acme-Flow-In-Realm_FS2_F	92
66	Acme-Flow-In-Src-Addr_FS2_F	93
67	Acme-Flow-In-Src-Port_FS2_F	94
68	Acme-Flow-In-Dst-Addr_FS2_F	95
69	Acme-Flow-In-Dst-Port_FS2_F	96
70	Acme-Flow-Out-Realm_FS2_F	97
71	Acme-Flow-Out-Src-Addr_FS2_F	98
72	Acme-Flow-Out-Src-Port_FS2_F	99
73	Acme-Flow-Out-Dst-Addr_FS2_F	100
74	Acme-Flow-Out-Dst-Port_FS2_F	101
75	Acme-Calling-RTCP-Packets-Lost_FS2	104
76	Acme-Calling-RTCP-Avg-Jitter_FS2	105
77	Acme-Calling-RTCP-Avg-Latency_FS2	106
78	Acme-Calling-RTCP-MaxJitter_FS2	107
79	Acme-Calling-RTCP-MaxLatency_FS2	108
80	Acme-Calling-RTP-Packets-Lost_FS2	109
81	Acme-Calling-RTP-Avg-Jitter_FS2	110
82	Acme-Calling-RTP-MaxJitter_FS2	111
83	Acme-Calling-Octets_FS2	102

CSV Placement	Attribute Name	VSA ID Number
84	Acme-Calling-Packets_FS2	103
85	Acme-FlowID_FS2_R	112
86	Acme-FlowType_FS2_R	113
87	Acme-Flow-In-REALM_FS2_R	114
88	Acme-Flow-In-Src-Addr_FS2_R	115
89	Acme-Flow-In-Src-Port_FS2_R	116
90	Acme-Flow-In-Dst-Addr_FS2_R	117
91	Acme-Flow-In-Dst-Port_FS2_R	118
92	Acme-Flow-Out-REALM_FS2_R	119
93	Acme-Flow-Out-Src-Addr_FS2_R	120
94	Acme-Flow-Out-Src-Port_FS2_R	121
95	Acme-Flow-Out-Dst-Addr_FS2_R	122
96	Acme-Flow-Out-Dst-Port_FS2_R	123
97	Acme-Called-RTCP-Packets-Lost_FS2	126
98	Acme-Called-RTCP-Avg-Jitter_FS2	127
99	Acme-Called-RTCP-Avg-Latency_FS2	128
100	Acme-Called-RTCP-MaxJitter_FS2	129
101	Acme-Called-RTCP-MaxLatency_FS2	130
102	Acme-Called-RTP-Packets-Lost_FS2	131
103	Acme-Called-RTP-Avg-Jitter_FS2	132
104	Acme-Called-RTP-MaxJitter_FS2	133
105	Acme-Called-Octets_FS2	124
106	Acme-Called-Packets_FS2	125
107	Acme-Session-Charging-Vector	54
108	Acme-Session-Charging-Function-Address	55
109	Acme-Firmware-Version	56
110	Acme-Local-Time-Zone	57
111	Acme-Post-Dial-Delay	58
112	Acme-Primary-Routing-Number	64
113	Acme-Originating-Trunk-Group	65
114	Acme-Terminating-Trunk-Group	66
115	Acme-Originating-Trunk-Context	67
116	Acme-Terminating-Trunk-Context	68

CSV Placement	Attribute Name	VSA ID Number
117	Acme-P-Asserted-ID	69
118	Acme-Ingress-Local-Addr	74
119	Acme-Ingress-Remote-Addr	75
120	Acme-Egress-Local-Addr	76
121	Acme-Egress-Remote-Addr	77
122	Acme-SIP-Diversion	70
123	Acme-Session-Disposition	60
124	Acme-Disconnect-Initiator	61
125	Acme-Disconnect-Cause	62
126	Acme-SIP-Status	71
127	Acme-Egress-Final-Routing-Number	134
128	Acme-Session-Ingress-RPH	135
129	Acme-Session-Egress-RPH	136

Port Mapping

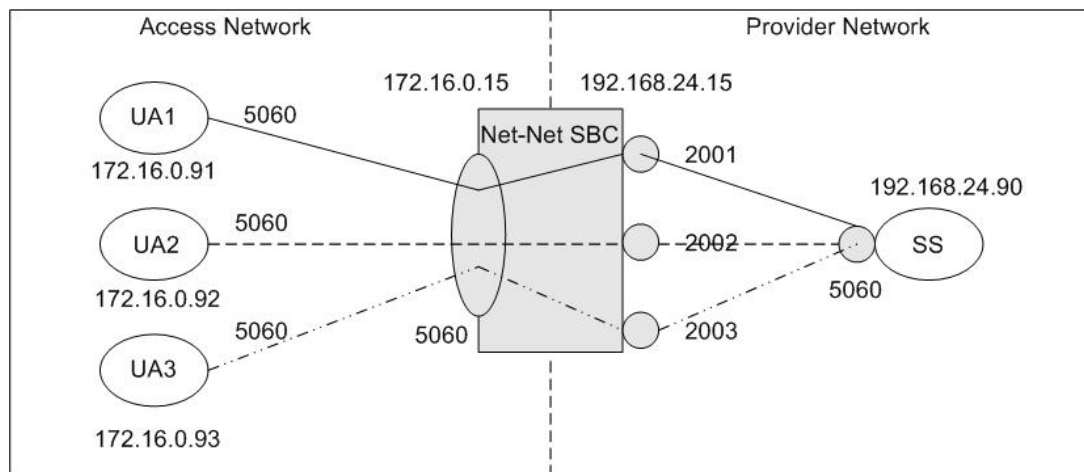
This section contains information about the SIP port mapping feature. SIP port mapping lets you allocate a unique SIP signaling transport address (IP address and UDP port) on the Net-Net SBC in the provider network for each registered endpoint (user agent).

About SIP Port Mapping

You might need to provide a unique signaling transport address for each registered endpoint for admission control; if required by your softswitch vendor. If you have questions about your softswitch, contact the vendor for assistance.

When a Net-Net SBC resides between the endpoints and the softswitch, the softswitch sees the same transport address (that of the Net-Net SBC) for all endpoints. By allocating a unique UDP port for each endpoint, the Net-Net SBC provides each of them a unique transport address.

The following example illustrates the SIP port mapping feature.



The diagram shows UA1, UA2, and UA3 are endpoints within the access network and that the SIP interface for the access network is 172.16.0.15:5060. On the provider network, the SIP interface is at 192.168.24.15, with the SIP port mapping feature enabled. The softswitch/registrar is also located on the provider network at 192.168.24.90:5060.

The diagram shows that port 2001 on the provider network is allocated to UA1 on the access network, port 2002 is allocated to UA2, and port 2003 is allocated to UA3. Because of this allocation, all SIP signaling messages sent from the endpoints in the access network to the softswitch on the provider network travel through an allocated signaling port. For example, all signaling messages between UA1 and the softswitch use 192.168.24.15:2001 as the transport address.

How SIP Port Mapping Works

The Net-Net SBC allocates SIP port mapping (signaling) ports during a REGISTER request that has registration caching applied. When you define a range of signaling ports for the SIP interface, you create a pool of signaling ports that can be allocated during the REGISTER request.

The Net-Net SBC allocates a signaling port from the pool when it creates the registration cache entry for a Contact in a REGISTER request. It allocates a separate signaling port for each unique Contact URI from the access side. The registration cache Contact entry contains the mapping between the Contact URI in the access/endpoint realm (the UA-Contact) and the Contact URI in the registrar/softswitch realm (the SD-Contact).

The SD-Contact is the allocated signaling port. The signaling port gets returned to the pool when the Contact is removed from the registration cache. The removal can occur when the cache entry expires; or when the endpoint sends a REGISTER request to explicitly remove the Contact from the registrar. When a signaling port returns to the pool it gets placed at the end of pool list; in a least-recently-used allocation method for signaling ports.

When the Net-Net SBC forwards the REGISTER request to the softswitch, it replaces the UA-Contact with SD-Contact. For example, if UA1 sends a REGISTER request with a Contact URI of `sip:ua1@172.16.0.91:5060`, it is replaced with `sip:192.168.24.15:2001` when the REGISTER request is forwarded to the registrar.

The same translation occurs when UA1 sends that same URI in the Contact header of other SIP messages. SIP requests addressed to the allocated signaling transport address (SD-Contact) are translated and forwarded to the registered endpoint contact address (UA-Contact).

Note: The maximum number of registered endpoints cannot exceed the number of signaling ports available. If no signaling ports are available for a new registration, the REGISTER request receives a 503 response.

The Net-Net SBC still processes requests received on the configured SIP port address. Requests sent into the registrar/softswitch realm that are not associated with a registered user will use the configured SIP port address.

About NAT Table ACL Entries

To enable SIP signaling messages to reach the host processor, the Net-Net SBC adds NAT table ACL entries for each SIP interface. With UDP without SIP port mapping applied, it adds a single ACL entry for each SIP port in the SIP interface configuration. For example:

untrusted entries:

intf:vlan index	source-ip/mask:port/mask	dest-ip/mask:port/mask	prot	type	
0/0:0	0.0.0.0	172.16.1.15:5060	UDP	static	10
0/3:0	0.0.0.0	192.168.24.15:5060	UDP	static	16
0/1:0	0.0.0.0	192.168.50.25:5060	UDP	static	17

Using SIP Port Mapping

When you use SIP port mapping, one or more ACL entries are added to the NAT table to enable the range of ports defined. The NAT table does not support the specification of port ranges. However, it does support masking the port to enable ranges that fall on bit boundaries. For example, an entry for `192.168.24.15:4096/12` defines the port range of `4096` through `8191`.

The algorithm for determining the set of ACLs for the port map range balances the need to represent the range as closely as possible, with the need to minimize the number of ACL entries. For example, a range of `30000` through `39999` would result in the following set of ACLs.

untrusted entries:

intf:vlan	source-ip/mask:port/mask	dest-ip/mask:port/mask	prot	type	index
0/3:0	0.0.0.0	192.168.24.15:30000/4	UDP	static	13
0/3:0	0.0.0.0	192.168.24.15:32768/4	UDP	static	14
0/3:0	0.0.0.0	192.168.24.15:36864/4	UDP	static	15

However, the first entry actually enables ports `28672` through `32767` and the last entry allows port `36864` through `40959`. If SIP messages are received on ports outside the configured range (`28672` through `29999` or `40000` through `40959` in this case), they are ignored.

Acme Packet recommends you use port map ranges that fall on bit boundaries to ensure the fewest possible ACL entries are created and only the configured ports are allowed by the ACLs. For example, a range of `32768` to `49151` provides for 16,384 signaling ports in a single ACL entry (`192.168.24.15:32768/2`).

Note: If the ACLs added for the port map range do not include the SIP port configured in the SIP interface; the normal SIP ACL entry for the SIP port is also added.

Dynamic Configuration

Dynamic configuration of SIP port mapping can cause disruption in service for existing registration cache entries; depending on the changes made to the defined port map range. If the range of mapping ports is reduced, it is possible that SIP signaling messages from the registrar/softswitch realm will no longer be sent to the host processor because of the changes in the NAT Table ACL entries.

When the range of mapping ports is changed, any signaling ports in the free signaling port pool not allocated to a registration cache entry are removed from the pool. When an allocated signaling port that is no longer part of the defined mapping port range is released, it is not returned to the pool of free steering ports.

The administrator is warned when the changed configuration is activated after the port map range of a SIP interface has been changed.

Registration Statistics

The SIP registration cache statistics include counters for free and allocated signaling ports. You can issue a show registration command to display the statistics:

17:36:55-190

SIP Registrations	-- Period --			----- Lifetime -----		
	Active	High	Total	Total	PerMax	High
User Entries	4	4	0	7	4	4
Local Contacts	4	4	0	7	4	4
Free Map Ports	12284	12284	0	12291	12288	12288
Used Map Ports	4	4	0	7	4	4
Forwards	-	-	1	22	4	
Refreshes	-	-	3	43	3	
Rejects	-	-	0	0	0	
Timeouts	-	-	0	1	1	
Fwd Postponed	-	-	0	0	0	
Fwd Rejected	-	-	0	0	0	
Refr Extension	0	0	0	0	0	0
Refresh Extended	-	-	0	0	0	

The labels for the first two items reflect the restructured registration cache:

- **User Entries:** counts the number of unique SIP addresses of record in the cache. Each unique address of record represents a SIP user (or subscriber). The address of record is taken from the To header in the REGISTER request. There might be one or more registered contacts for each SIP user. The contacts come from the Contact header of the REGISTER request.
- **Local Contacts:** counts the number of contact entries in the cache. Because the same user can register from multiple endpoints (user agents); the number of Local Contacts might be higher than the number of User Entries.
- **Free Map Ports:** counts the number of ports available in the free signaling port pool.
- **Used Map Ports:** counts the number of signaling ports allocated for registration cache entries. The value of Used Map Ports will equal the number of Local Contacts when the port mapping feature is used for all registrar/softswitch realms in the Net-Net SBC.

Configuring SIP Port Mapping

You configure the SIP port mapping feature on a per-realm basis. Configure the port map range on the SIP interface for the realm where the registrar/softswitch resides. Port mapping is only applied when the access/ingress realm has registration caching and/or HNT enabled.

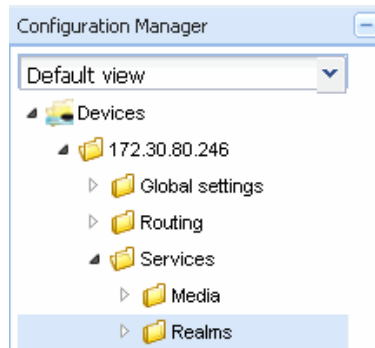
The range of SIP mapping ports must not overlap the following:

- Configured SIP port, which might be used for signaling messages not associated with a registered endpoint.
- Port range defined for steering pool configuration using the same IP address as the SIP interface. If overlap occurs, the NAT table entry for the steering port used in a call prevents SIP messages from reaching the host processor.

You can configure SIP port mapping using Net-Net Central.

To configure SIP port mapping:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**. Scroll to the Port mapping area.
6. **Start port**—Enter the starting port for the range of SIP ports available for SIP port mapping. The valid range is 1025 through 65535. If you set the value to zero (0), SIP port mapping is disabled.
7. **End port**—Enter the ending port for the range of SIP ports available for SIP port mapping. The valid range is 1025 through 65535. If you set the value to zero (0), SIP port mapping is disabled.

Port Mapping

Start port

32768

End port

40959

8. Click **Apply**.

SIP Port Mapping for TCP and TLS

In releases prior to S-C6.2.0, the Net-Net SBC supports SIP port mapping for UDP and now you can enable this feature for SIP sessions using TCP and TLS. Port mapping enables the Net-Net SBC to allocate a unique port number for each endpoint registering through it by giving it a transport address (or hostport) in the registered Contact.

When you enable this feature for TCP and TLS, the Net-Net SBC designates a port from a configured range for each endpoint that registers with SIP servers in the SIP interface's realm. You establish that range of ports using the **port-map-start** and **port-map-end** parameters. Unlike its behavior with UDP port mapping—where the Net-Net SBC sends requests on the SIP interface from the allocated port mapping, the Net-Net SBC sends all requests over an existing connection to the target next hop for TCP/TLS port mapping. If a connection does not exist, the system creates one. So for TCP/TLS port mapping, only the Contact header contains the transport

address of the mapping port (i.e., the transport address of the configured SIP port). And the system refuses TCP and TLS connections on the allocated mapping port.

With TCP/TLS port mapping enabled, the Net-Net SBC sends the Path header with the transport address in Register requests, unless you specify that it should not do so. Standards-conformant SIP servers (that support RFC 3327) might attempt to send requests to the allocated mapping port if the Path header is absent.

Note: ACL entries in the NAT table that permit TCP/TLS signaling for a SIP port configuration with TCP/TLS port mapping are the same as they would be for a TCP/TLS SIP port without port mapping enabled. Additional ACL entries that need to be set up for UDP port mapping are not required for TCP/TLS port mapping.

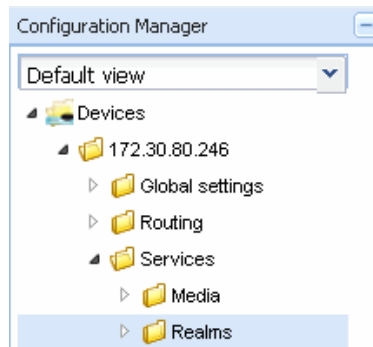
Configuring TCP/TLS Port Mapping

You enable TCP/TLS port mapping in a per-realm basis using the SIP interface configuration; setting the **tcp-port-mapping** value in the Options parameter enables the feature. Enabling this parameter turns on the port mapping feature for UDP as well.

By default, the Net-Net SBC includes the Path header in Register requests it sends from that SIP interface. If you do not this header to be included, however, you can set the value as **tcp-port-mapping=nopath**.

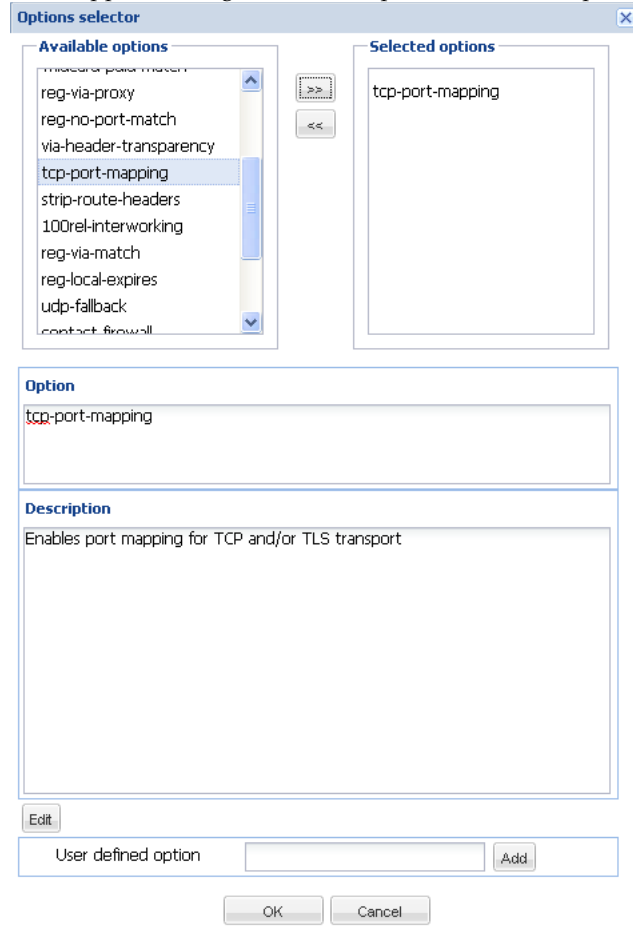
To enable TCP/TLS port mapping for a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to Options in SIP service.
7. Click **Options**. The Options dialog box appears in the display pane.
8. Click **tcp-port-mapping** in the Available options list to select it.

9. Click  to move the option to the Selected options list. The option name appears along with a description in the description section of the window.



10. Click **OK**.
11. Click **Apply**.

SIP Registration Via Proxy

The Net-Net SBC supports a number of features that require it to cache registration information for UAs (endpoints) registering and receiving requests through it. For those features to operate correctly, the Net-Net SBC must act as the outbound proxy through which these endpoints register.

In order to support deployments where a proxy sits between the Net-Net SBC and the endpoints, the Net-Net SBC must consider the bottom Via header it receives from endpoints when constructing and matching cache registration entries. And when you use SIP port mapping, the system must use the bottom Via header as a way to determine the endpoint uniquely so that it can have a unique mapping port when the SIP interface is configured with the reg-via-key=all option.

Using the reg-via-proxy option, you can enable your Net-Net SBC to support endpoints that register using an intervening proxy. You can set this option either for a realm or for a SIP interface. If you use it for a SIP interface, add to the SIP interface pointing toward the proxy and endpoints—the access side.

Considerations for Reg-Via-Key and Port Mapping

When you set the **reg-via-proxy** option, the Net-Net SBC includes the bottom Via header from received requests in the registration cache Via Key. The system also uses it for determining whether or not the request matches a registration cache entry. Each unique bottom Via received a unique mapping port when you turn SIP port mapping on and set the SIP interface with the **reg-via-key=all** option.

Request Routing

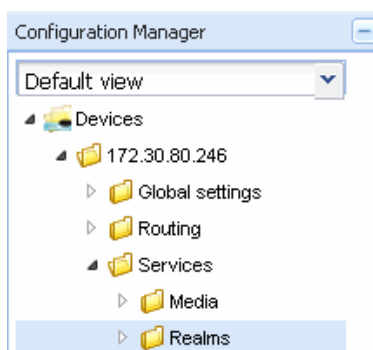
So that requests addressed to the corresponding registered contact are routed to the proxy, the Net-Net SBC includes the intervening proxy (i.e., the top Via) in the routing information for the registration cache when you set **reg-via-proxy**. To carry out this routing scheme, the system adds a Path header (if none is present) to the REGISTER. But it removes the Path header prior to sending the REGISTER to the registrar.

Note that when the received REGISTER contains a Path header, the Net-Net SBC uses it for routing requests toward the endpoint and includes it in the forwarded REGISTER request—as is the case when you do not enable SIP registration via proxy.

Configuring SIP Registration Via Proxy

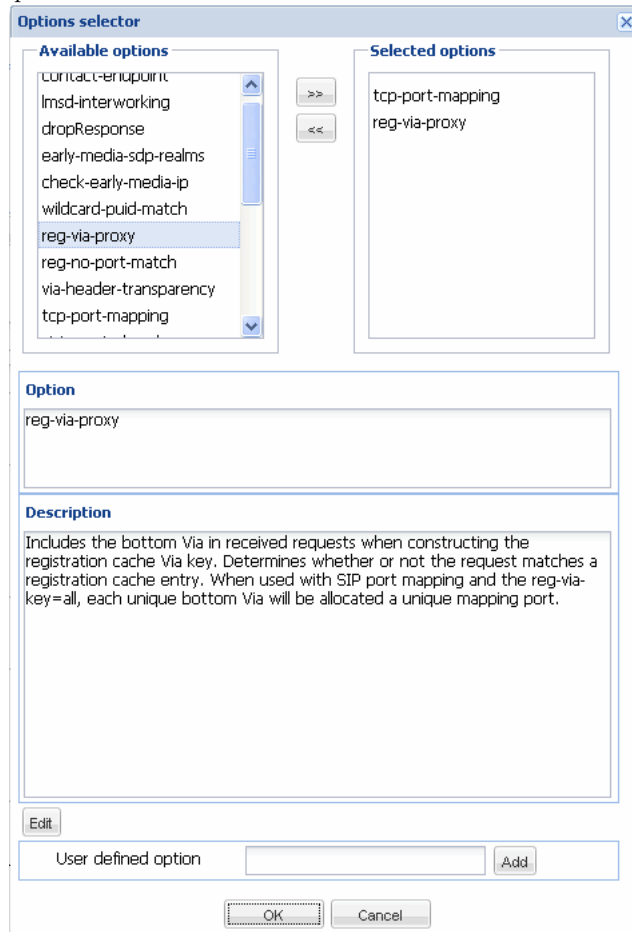
To configure SIP registration via proxy:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to Options in SIP service and click **Options**. The Options dialog box appears.
7. Click **reg-via-proxy** in the Available options list to select it.

8. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



9. Click OK.
10. Click Apply.

Configurable Route Recursion

When the Net-Net SBC routes SIP requests from a UAC to a UAS, it might determine that there are multiple routes to try based on a matching local policy. The Net-Net SBC recurses through the list of routes in a specific order according to your configuration and the quality of the match. There are other scenarios when a UAS replies with a 3xx Redirect response to the Net-Net SBC, the 3xx response can include multiple Contacts to which the request should be forwarded in a specific order. In both cases, the Net-Net SBC needs to recurse through a list of targets.

When the Net-Net SBC receives a non-successful (or non-6xx response) final response from the UAS, and there are multiple targets for the original request, the Net-Net SBC forwards the request to the next target and wait for a response.

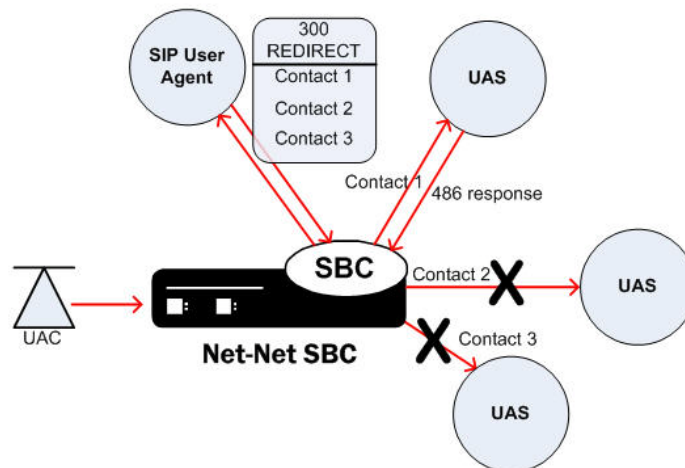
Use SIP route recursion when you want the Net-Net SBC to forward a response to the UAC and stop recursing through the target list immediately after receiving the 3xx, 4xx, or 5xx response code that you configure. When this feature is disabled, the Net-Net SBC only stops recursing when it receives a message with a 401 or 407

response code. With route recursion, you can configure a specific message or range of messages to stop recursing on when received. The Net-Net SBC retains its default behavior to stop recursing on a 401 or 407 response code when SIP Route Recursion is configured on a SIP interface. The Net-Net SBC will always stop recursing when it receives a global failure (6xx); this behavior is not configurable.

You can disable response recursion for either a SIP interface or for a SIP session agent, providing you with flexibility for various network architectures. For instance, a PSTN gateway might be the only hop to reach a given endpoint, whereas several session agents might need to be contacted if multiple devices map to a contacted address of record.

Example 1

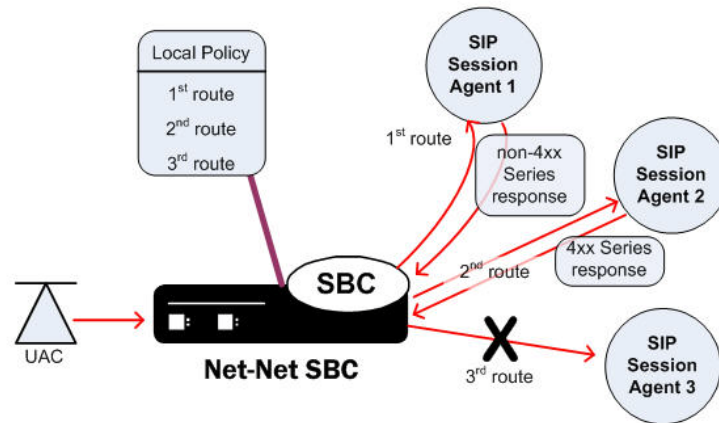
A softswitch might return a list of contacts for multiple PSTN gateways in a Redirect message. If the PSTN target number contacted on redirection is busy, a 486 response will be sent to the Net-Net SBC. Since the single target is located in the PSTN, a subsequent request through a different gateway will yield another 486 response. The Net-Net SBC should be configured to return the 486 response to the UAC immediately. No other SIP requests should be sent to applicable targets/contacts that were enumerated in the redirect list. See the following example:



Example 2

The Net-Net SBC might determine from a local policy lookup that several routes are applicable for forwarding a SIP message. The Net-Net SBC will try each route in turn, but the SIP response recursion disable feature can be implemented to stop the

route recursion when a configured responses message is received by the Net-Net SBC. See the following example:



There are a few conditions on the parameter used to configure response recursion:

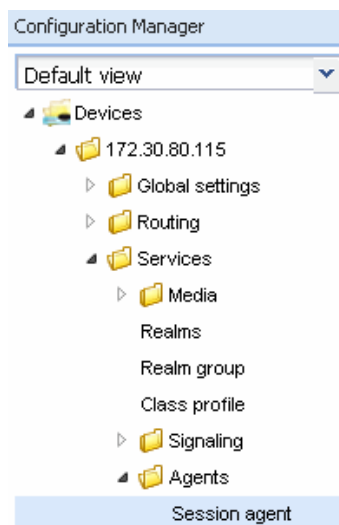
- SIP Route Recursion is configurable for either the SIP interface or session agent.
- 401 and 407 are preconfigured for all configured SIP interfaces. They are not configured for session agents.
- The format is a comma-separated list of response codes or response code ranges: 404, 484-486.
- Only response codes that fall within the 3xx, 4xx, and 5xx range may be specified.

Configuring Route Recursion

You can configure a session agent or a SIP interface for SIP route recursion using Net-Net Central.

To configure a session agent for SIP route recursion:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the Constraints tab at the top of the content area.

4. Scroll to the Response codes that stop recursion section, click **Add**. The SIP Response codes window appears.
5. **Response codes that stop recursion**—Enter the response code value. You can enter a single response code or response code range; or a comma-delimited list of response codes or response code ranges.

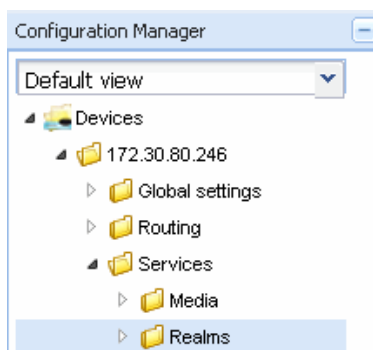
The 'Add' dialog box has a title bar with 'Add' and a close button. Inside, there is a label 'Response codes that stop recursion' and a text input field containing '484-486'. At the bottom are 'OK' and 'Cancel' buttons.

6. Click **OK** to close the dialog box. The response code value appears in the list of response codes for recursion.
7. Click **Apply**.

Configuring a SIP Interface for SIP Route Recursion

To configure a SIP interface for SIP route recursion:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to the List of response code for recursion section and click **Add**. The SIP response codes window appears.

List of response code for recursion

The window shows a text input field containing '401,407'. To the right of the field are two buttons: 'Add' (highlighted in blue) and 'Delete'.

7. **Response code**—Enter a single response code or response code range; or a comma-delimited list of response codes or response code ranges. Click **OK** to close the dialog box. The response code value appears in the list of response codes for recursion.
8. Click **Apply**.

Event Package Interoperability

Service providers often deploy a Net-Net SBC on the border of an access network, where it sits between the SIP endpoints (user agents) and the service provider's application server. The application server and the user agents sometimes use various SIP event packages to exchange and maintain state information. The SUBSCRIBE and NOTIFY methods are used to establish subscriptions to the event packages and to report state changes to the subscribing entity.

The SIP global contact option addresses interoperability in the Dialog and Presence event packages that are used in hosted PBX and IP Centrex offerings. State information is passed in the message body of a NOTIFY request; this message body is encoded in an XML format described by the Content-Type header. The Net-Net SBC needs to update certain fields in the body to account for dialog mapping and SIP NAT functionality between the access and service provider realms. Often the subscriptions are established using URIs learned from Contact headers in the user agent registrations or dialog establishment (INVITE/SUBSCRIBE). For this, a Net-Net SBC requires a Contact URI that is usable and routable outside of an existing dialog.

The SIP global contact option enables persistent URIs in the Contact headers inserted into outgoing SIP messages. If this option is not used, URIs placed in the Contact header of outgoing messages are only valid within the context of the dialog to which the message is associated.

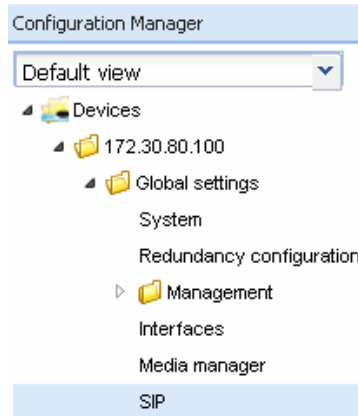
RFCs associated with this feature are:

- A. B. Roach, "Session Initiation Protocol (SIP)-Specific Event Notification," RFC 3265, June 2002
- J. Rosenberg, "A Presence Event Package for the Session Initiation Protocol (SIP)," RFC 3856, August 2004
- J. Rosenberg, et al. "Data Format for Presence Using XML," <http://www.iptel.org/info/players/ietf/presence/outdated/draft-rosenberg-impp-pidf-00.txt>, Work In Progress (expired), June 2000
- J. Rosenberg, H. Schulzrinne, R. Mahy, "An INVITE Initiated Dialog Event Package for the Session Initiation Protocol (SIP)," draft-ietf-sipping-dialog-package-06.txt, Work In Progress, April 2005
- H. Sugano, et al., "Presence Information Data Format (PIDF)," RFC 3863, August 2004

Configuring Event Package Interoperability

You can configure SIP event package interoperability using Net-Net Central.

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to the Options section and click **Options**. The Options dialog box appears.
5. Click **global-contact** in the Available options list to select it.



6. Click **>>** to move the option to the opposite list. The option name appears along with a description in the lower part of the window. Click **OK**.
7. Click **Apply**.

REGISTER Forwarding After Call-ID Change

Sometimes an endpoint reboots and performs a third party registration before its old registration expires. During this re-registration, the Contact header is the same as it was pre-reregistration. As a consequence of the reboot, the SIP Call-ID changes. In this situation, the Net-Net SBC does not forward the REGISTER to the registrar, because it believes the endpoint is already registered, based on a previous registration from the same Contact: header URI.

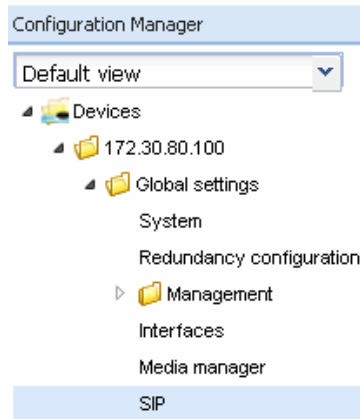
To remedy this problem, the Net-Net SBC keeps track of the Call-ID in its registration cache. You can configure an option that forces the Net-Net SBC to

forward a REGISTER message to the registrar when the Call-ID header changes in a REGISTER message received from a UAC that is re-registering.

Configuring REGISTER Forwarding After Call-ID Change

To configure SIP REGISTER forwarding after a Call-ID change:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to the Options section and click **Options**. The Options dialog box appears.
5. Click **forward-reg-callid-change** in the Available options list to select it.
6. Click  to move the option to the opposite list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

Local Response Code Mapping

SIP local response code mapping (known as the response code translation profile in Net-Net Central) lets you establish a table that maps SIP response-received messages to response-to-send messages.

SIP local response code mapping is used with the SIP responses generated by the Net-Net SBC toward a specific SIP session agent. You provision the mapping of the response codes used by the Net-Net SBC when it generates the responses towards a session agent.

You create the SIP local response code map using the mapping functionality and then assigning that map to a session agent.

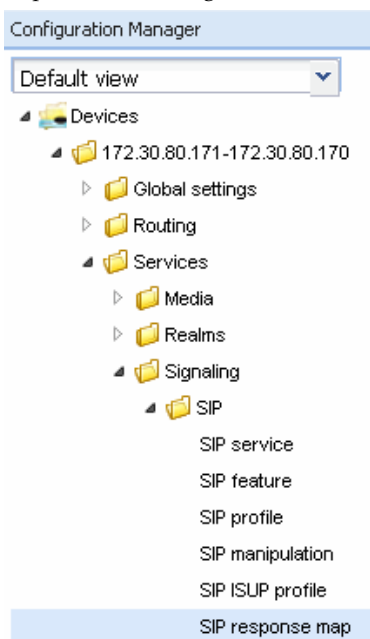
Note: The configured response map is not used when the Net-Net SBC is acting as proxy for the responses to this session agent.

Configuring Local Response Code Mapping

This section explains how to create the SIP response code map and how to assign it to a specific session agent using Net-Net Central.

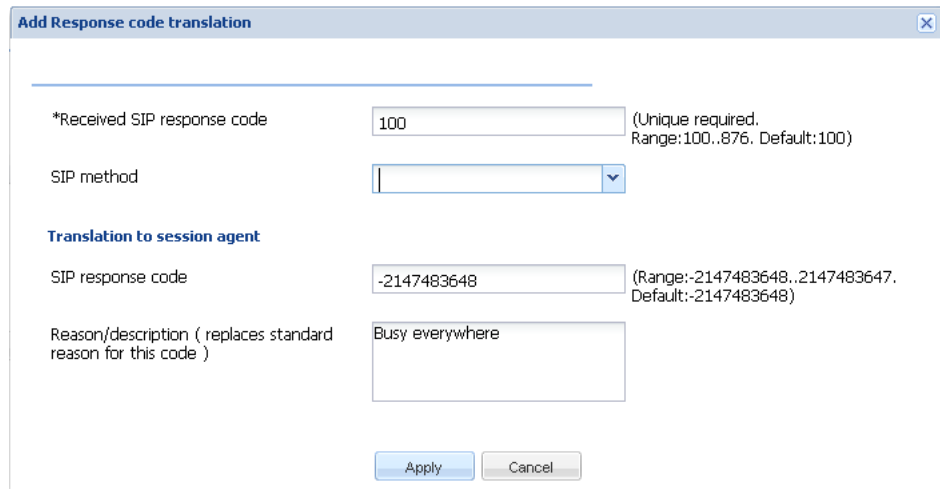
To create a SIP response code map for a session agent:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP response map. The SIP response map table appears in the content area.
3. Click **Add**. The Add SIP response code translation window appears.
4. **SIP response code translation profile Name**—Enter the name for the response code translation profile and click **OK**. The new profile appears in the Response map table.
5. Click the new response code profile in the table and click **Edit**.
6. Click **Add**. The Add SIP response code translation dialog box appears.
7. **Received SIP response code**—Enter the original SIP response code as an integer from 100 to 699.
8. **SIP response code**—Enter the SIP response code into which you want the original response code to be translated. Use an integer from 100 to 699.

9. **Reason/description (replaces standard reason for this code)**—Enter a reason for the translated code. This response comment is sent with the translated code.



The dialog box titled "Add Response code translation" contains the following fields:

- *Received SIP response code: 100 (Unique required. Range:100..876. Default:100)
- SIP method: (empty dropdown)
- Translation to session agent:
 - SIP response code: -2147483648 (Range:-2147483648..2147483647. Default:-2147483648)
 - Reason/description (replaces standard reason for this code): Busy everywhere

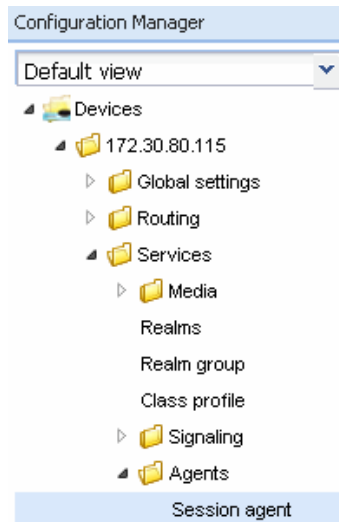
Buttons: Apply, Cancel

10. Click **Apply** to save the values and close the Add SIP response code translation profiles window. You return to the Response map tab displaying the Response map table.

Assigning the Response Code Map to a Session Agent

To assign a response code map

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent in the table and click **Edit**.
4. Scroll to the SIP message handling section.
5. **Local response code translation profile**—Choose the local response code translation profile from the combination box.

Local response code translation profile

6. Click **Apply**.

Session Agent Ping Message Formatting

You can configure the user portion of the From: header, the To: header, and the Request-URI in the ping-type message that the Net-Net SBC sends to a session agent. This feature is required for interoperability with certain E911 servers.

In the following example of a session agent ping-type message, you can set the user portion of the Request-URI (the text bob in the OPTIONS method line) and the user portion of the From: header (the text bob in the From: header) to the same new value. You can also set the user portion of the To: header (the text anna in the To: header) to its own new value.

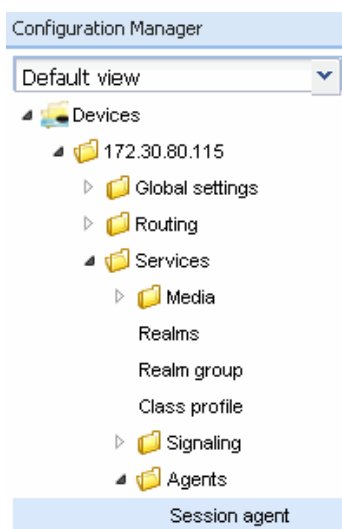
```
OPTIONS sip:bob@sip.com SIP/2.0
From: UA1 <sip:bob@sip.com>
To: NUT <sip:anna@gw.sip.com>
Call-ID: 123abc@desk.sip.com
CSeq: 1 OPTIONS
Contact: <sip:UA1@client.sip.com>
Accept: application/sdp
Content-Length: 0
```

If you do not enable this feature, the session agent ping-type message will contain the text “ping” in all cases.

Configuring Session Agent Ping Message Formatting

To configure session agent ping message formatting:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent to which you want to use session agent ping message formatting in the table and click **Edit**.
4. Scroll to the Ping parameters section.
5. **User part to be used in PING To header**—Enter the user name.

6. **User part to be used in PING From header**—Enter the user and user URI portion.

User part to be used in PING To header	anna
User part to be used in PING From header	bob

7. Click **Apply** to continue.

PAI Stripping

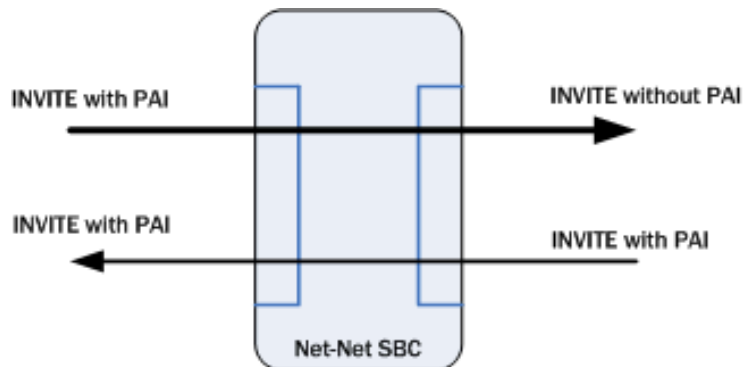
The Net-Net SBC can strip P-Asserted-Identity (PAI) headers so that service providers can ensure an extra measure of security against malicious users pretending to be legitimate users. To pretend to represent another account, the malicious users simply send an INVITE with an imitation PAI. This feature allows real-time detection of such fraudulent use.

This feature uses a combination of:

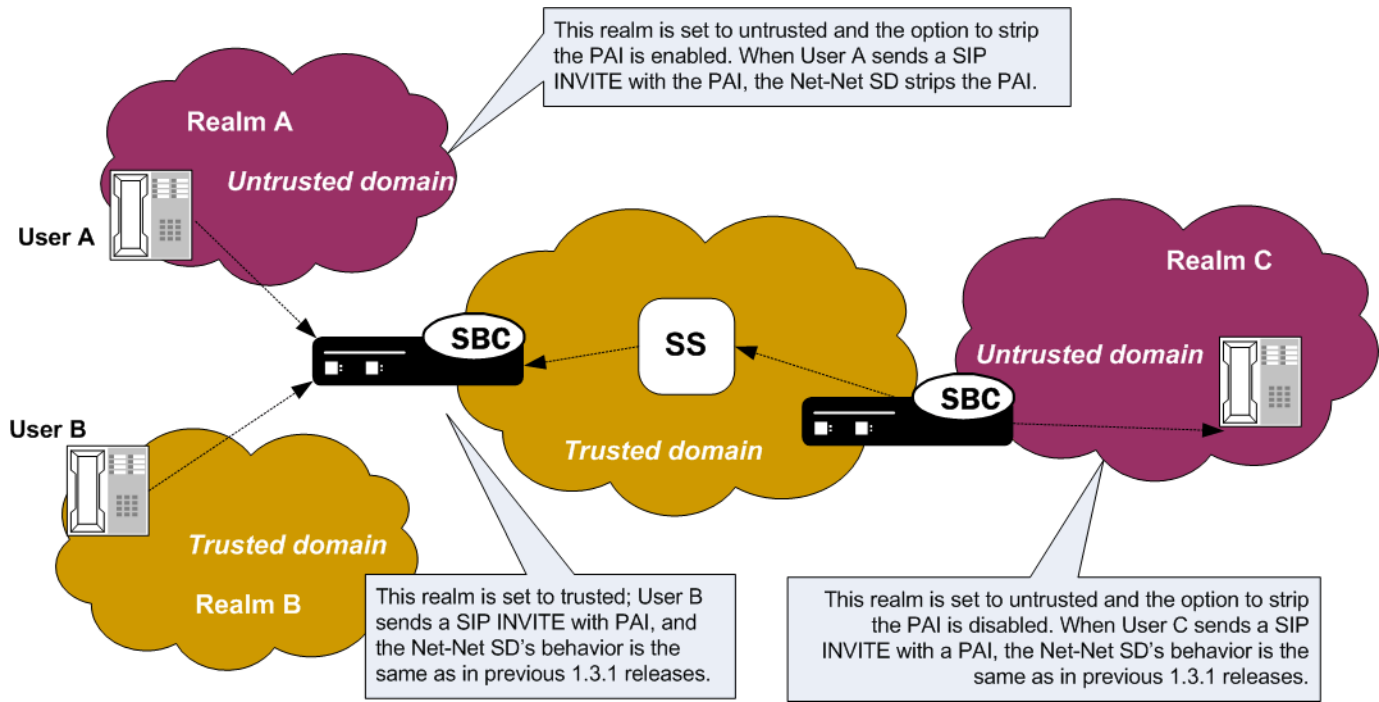
- DoS protection applied on a per-realm basis
- SIP PAI header stripping

The combination of these settings can produce different results for the SIP PAI stripping feature.

- *SIP PAI header stripping enabled for an untrusted realm*—If the PAI stripping parameter is set to enabled in a realm that is untrusted, then the Net-Net SBC strips the PAI headers from SIP INVITEs that are received from the external address, regardless of the privacy type. The Net-Net SBC then sends the modified INVITE (without the PAI). If the INVITE comes from a trusted realm, then the Net-Net SBC does not strip the PAI header and the Net-Net SBC behaves as it does when you are using previous 1.3.1 releases.



- **Multiple SIP PAIs in a SIP INVITE**—The Net-Net SBC removes all PAIs when there are multiple PAIs set in SIP INVITEs that come from untrusted realms.
- **Net-Net SBC behavior bridging trusted and untrusted realms**—The following graphics shows you how Net-Net SBCs can be positioned and configured to handle PAI stripping between trusted and untrusted realms.

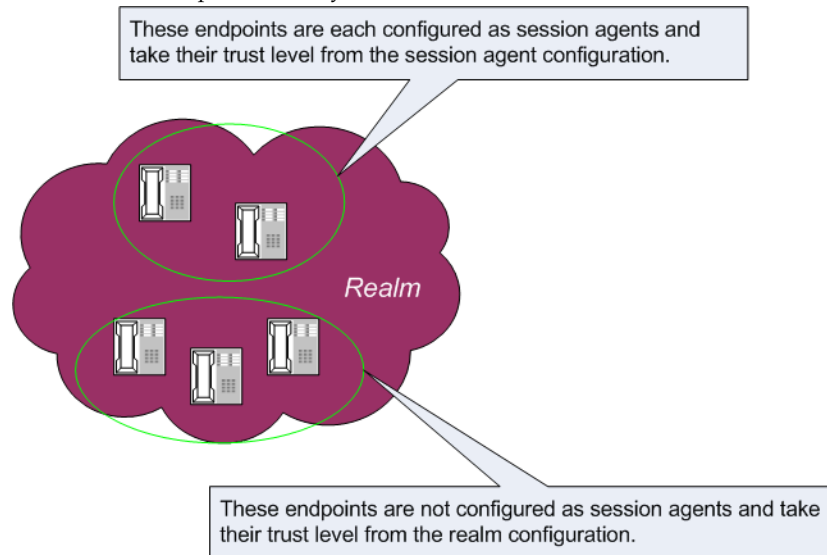


Realm Configuration Settings	REALM A	REALM B	REALM C
Realm designation trusted or untrusted (trust-me)	Disabled	Enabled	Enabled
SIP PAI stripping (pai-strip)	Enabled	Enabled or disabled	Disabled
Net-Net SBC's behavior	Strip PAI regardless of privacy type	Same as behavior for SIP privacy support in previous 1.3.1 releases	Same as behavior for SIP privacy support in previous 1.3.1 releases

SIP PAI Stripping Configuration

Note how the Net-Net SBC behaves when you combine the designation of a realm as trusted/untrusted and SIP PAI stripping is enabled.

Trust is also established in the session agent configuration, and that the trust level set in a session agent configuration overrides the trust set in a realm configuration. For example, a realm might have several endpoints, some of which are associated with session agents and some of which are not. The endpoints that have configured session agent will take their trust level from the session agent parameters you set; the other endpoints, ones that are not associated with session agents, take their trust level from the realm parameters you set.



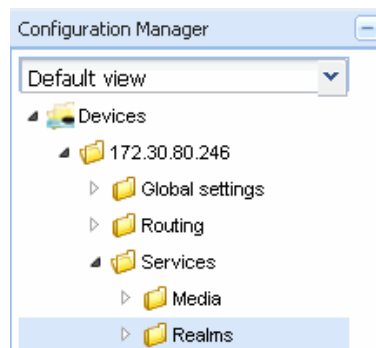
Take this relationship into consideration when you configure SIP PAI header stripping, or it will not work as designed.

For the sample configuration cited below, the desired Net-Net SBC behavior is to always strip the PAI regardless of privacy type.

Configuring SIP PAI Stripping

To configure SIP PAI stripping:

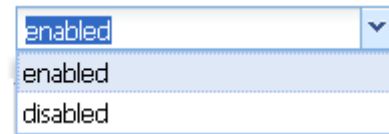
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm to which you want to apply SIP PAI stripping in the table and click **Edit**.

4. **Strip P-Asserted-Identified (PAI) headers from incoming requests**—Select enabled from the combination box to activate it.

Strip P-Asserted-Identified (PAI) headers
from incoming requests



enabled

enabled

disabled

5. Click **Apply**.

SIP Statuses to Q.850 Reasons Mapping

This section explains the Net-Net SBC's ability to map Q.850 cause values with SIP responses, which is used in SIP calls and calls that require IWF.

RFC 3326 defines a header that might be included in any in-dialogue request. This reason header includes cause values that are defined as either a SIP response code or ITU-T Q.850 cause values. You can configure the Net-Net SBC to support sending and receiving RFC 3326 in SIP messages for:

- Mapping H.323 Q.850 cause values to SIP responses with reason header and cause value
- Mapping SIP response messages and RFC 3326 reason header and cause
- Locally generated SIP response with RFC 3326 reason header and cause

As specified in RFC 3326, the Net-Net SBC sends SIP responses to the softswitch that contain the received Q.850 cause code and the reason.

Though the Net-Net SBC can generate RFC 3326 headers, the default behavior for this feature is disabled. Furthermore, the Net-Net SBC can receive and pass SIP error messages (4xx, 5xx, and 6xx) that contain the SIP reason header with a Q.850 cause code and reason (as specified in RFC 3326). If the Net-Net SBC receives an error message without the Reason header, then the Net-Net SBC is not required to insert one.

In calls that require IWF, the Q.850 cause generated in the SIP response are the same as the cause received in the following H.225 messages: Disconnect, Progress, Release, Release Complete, Resume Reject, Status, and Suspend Reject. In addition, the Q.850 cause codes that the Net-Net SBC receives in RFC 3326 headers are passed to the H.323 part of the call unmodified; the H.323 call leg uses this cause code for releasing the call.

SIP to SIP Calls

The SIP Reason header might appear in any request within a dialog, in a CANCEL request, and in any response where the status code explicitly allows the presence of this header field. The syntax of the header follows the standard SIP parameter:

```
Reason: SIP;cause=200;text="completed elsewhere"
Reason: Q.850;cause=16;text="Terminated"
```

This feature attends to the following possible SIP call scenarios:

- When the Net-Net SBC receives a SIP request or SIP response that contains the Reason header, the Net-Net SBC passes it without modification.
- When it generates a SIP response, the Net-Net SBC includes the RFC 3326 Reason header containing a Q.850 cause code and reason. This is the case for all local conditions and for all internally generated error responses (4xx, 5xx, and 6xx) to an initial SIP INVITE.

Possible local error scenarios are:

- invalid-message
- cpu-overloaded
- media-released
- media-not-allocated

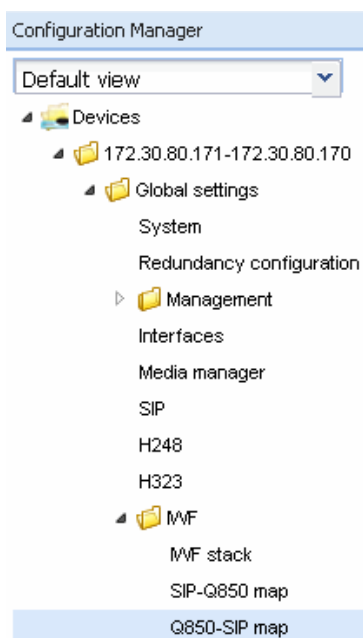
Configuring SIP Statuses to Q.850 Reasons Mapping

You can create a local response map and enable the Net-Net SBC to use the Reason header.

Creating a Local Response Map

To create a local response map:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click SIP-Q850 Map.
3. Click **Add** to add the configuration. The SIP > H.323 calls table appears in the content area.
4. Click **Add**. The Add SIP Q.850 Map Entry dialog box appears.
5. **SIP status code**—Set the SIP response code that you want to map to a particular Q.850 cause code and reason. The valid range for values is 100-699.
6. **Q850 cause code**—Set the Q.850 cause code that you want to map to the SIP response code that you set in step 5.
7. **Q850 reason**—Set the Q.850 reason corresponding to the Q.850 cause code. There is no default. If your value has spaces between characters, then your entry must be surrounded by quotation marks.

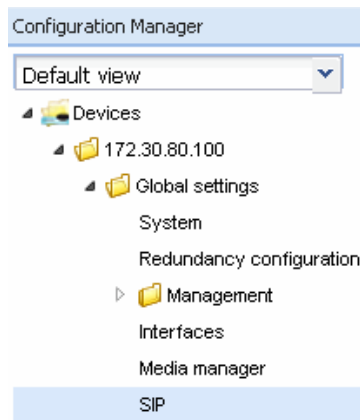
8. Click **Apply**. The entry appears in the SIP->H.323 calls table.
9. Repeat this process to create the number of local response map entries that you need.

- Click **Apply**.

Enabling the Net-Net SBC to Add the Reason Header

To enable the Net-Net SBC to add the reason header:

- Expand the Configuration Manager slider > Global settings.



- Click SIP.
- Click **Add** to add the SIP configuration in the content area.
- Reason headers in SIP responses**—Select enabled from the combination box.

Reason headers in SIP responses

- Click **Apply**.

Calls Requiring IWF

For interworking calls between SIP and H.323, you can configure mappings for:

- SIP status codes to Q.850 values
- Specific Q.850 cause codes to SIP status codes

If it cannot find the appropriate mapping, then the Net-Net SBC uses default mappings defined in the Default Mappings table below.

The following describes how the Net-Net SBC handles different IWF call scenarios:

- SIP request containing a Reason header—When it receives a request containing a Reason header, the Net-Net SBC determines if the request is a SIP BYE or SIP CANCEL message. RFC 3326 states that the Reason header is mainly used for these types of requests. If there is a Reason header and it contains the Q.850 cause value, then the Net-Net SBC releases the call on the H.323 side using the specified cause value.
- SIP response—When it receives the error response to an initial SIP INVITE, the Net-Net SBC uses its SIP-Q.850 map to determine the Q.850 that it will use to release the call. If there is not a map entry, then the Net-Net SBC uses the default mappings shown in the Default Mappings table.
- Active call released from the H.323 side—If an active call is released from the H.323 side, the Net-Net SBC checks the outgoing realm (the SIP side) to see if the addition of the Reason header is enabled. If it is, then the Net-Net SBC adds the Reason header in the SIP BYE request with the Q.850 value it received from the H.323 side.

- Error during setup of the call on the H.323 side—In the event of an error during setup on the H.323 side of the call, the Net-Net SBC needs to send:
 - An error response, if this is a SIP to H.323 call
 - A SIP CANCEL, if this is a H.323 to SIP call and the H.323 side hangs up before the call is answered on the SIP side

In this case, the Net-Net SBC checks to see if adding the Reason header is enabled in the IWF configuration. If it is, then the Net-Net SBC adds the Reason header with the Q.850 cause value it received from the H.323 side.

- Call released due to a Net-Net SBC error—If the call is released due a Net-Net SBC error and adding the Reason header is enabled in the IWF configuration, the error response to the initial INVITE contains the Reason header. The Net-Net SBC checks the SIP to Q.850 map configurations to determine whether or not the SIP error response code it is generating is configured. If it is, then the Net-Net SBC maps according to the configuration. If it is not, the Net-Net SBC derives cause mapping from the default table.

Like the configuration for SIP-only calls that enable this feature, you can set a parameter in the IWF configuration that enables adding the Reason header in the SIP requests or responses.

Default Mappings

This table defines the default mappings the Net-Net SBC uses when it cannot locate an appropriate entry that you have configured.

Q.850 Cause Value		SIP Status		Comments
1	Unallocated number	404	Not found	
2	No route to specified transit network	404	Not found	
3	No route destination	404	Not found	
16	Normal calling clearing		BYE message	A call clearing BYE message containing cause value 16 normally results in the sending of a SIP BYE or CANCEL request. However, if a SIP response is to be sent to the INVITE request, the default response code should be used.
17	User busy	486	Busy here	
18	No user responding	408	Request timeout	
19	No answer from the user	480	Temporarily unavailable	
20	Subscriber absent	480	Temporarily unavailable	

Q.850 Cause Value		SIP Status		Comments
21	Call rejected	603	Decline (if location filed in Cause information element indicates user; otherwise 403 Forbidden is used)	
22	Number changed	301	Moved permanently (if information in diagnostic field of Cause information element is suitable for generating SIP Contact header; otherwise 410 Gone is used)	
23	Redirection to new destination	410	Gone	
25	Exchange routing error	483	Too many hops	
27	Destination out of order	502	Bad gateway	
28	Address incomplete	484	Address incomplete	
29	Facility rejected	501	Not implemented	
31	Normal, unspecified	480	Temporarily unavailable	
34	No circuit, channel unavailable	503	Service unavailable	
38	Network out of order	503	Service unavailable	
41	Temporary failure	503	Service unavailable	
42	Switching equipment congestion	503	Service unavailable	
47	Resource unavailable unspecified	503	Service unavailable	
55	Incoming calls barred with CUG	403	Forbidden	
57	Bearer capability not authorized	403	Forbidden	
58	Bearer capability not presently available	503	Service unavailable	
65	Bearer capability not implemented	488	Not acceptable here	
69	Requested facility not implemented	501	Not implemented	

Q.850 Cause Value		SIP Status		Comments
70	Only restricted digital information available	488	Not acceptable here	
79	Service or option not implemented, unspecified	501	Not implemented	
87	User not member of CUG	403	Forbidden	
88	Incompatible destination	503	Service unavailable	
102	Recovery on timer expiry	504	Server time-out	

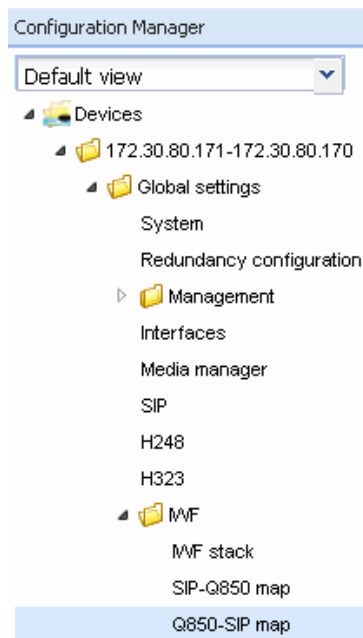
Configuring Calls Requiring IWF

You can configure mappings for SIP status codes to Q.850 values and mappings for particular Q.850 cause codes to SIP status codes using Net-Net Central.

SIP to H.323 Calls

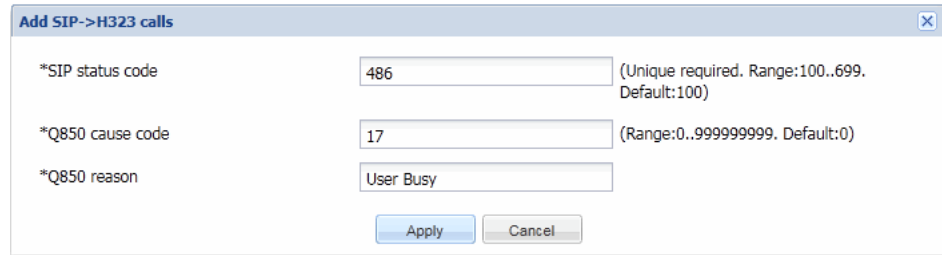
To configure a SIP status to Q.850 reason with cause mapping:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click SIP-Q850 Map.
3. Click **Add** to add the configuration. The SIP > H.323 calls table appears in the content area.
4. Click **Add**. The Add SIP Q.850 Map Entry dialog box appears.
5. **SIP status code**—Set the SIP response code that you want to map to a particular Q.850 cause code and reason. The valid range for values is 100-699.

6. **Q850 cause code**—Set the Q.850 cause code that you want to map to the SIP response code that you set in step 5.
7. **Q850 reason**—Set the Q.850 reason corresponding to the Q.850 cause code. There is no default. If your value has spaces between characters, then your entry must be surrounded by quotation marks.



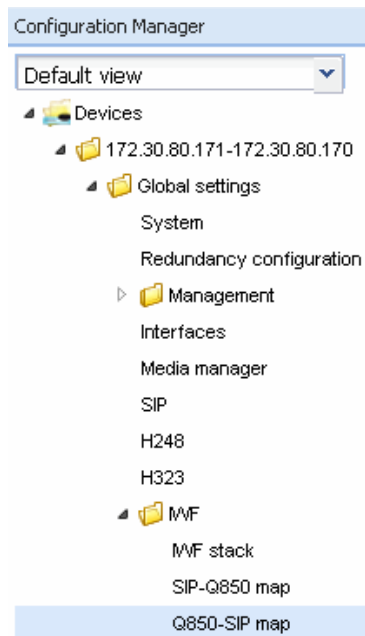
The dialog box titled "Add SIP->H323 calls" contains three input fields and two buttons. The first field, labeled "*SIP status code", has the value "486" and a tooltip "(Unique required. Range:100..699. Default:100)". The second field, labeled "*Q850 cause code", has the value "17" and a tooltip "(Range:0..999999999. Default:0)". The third field, labeled "*Q850 reason", has the value "User Busy". At the bottom are "Apply" and "Cancel" buttons.

8. Click **Apply**. The entry appears in the SIP->H.323 calls table.
9. Repeat this process to create the number of local response map entries that you need.
10. Click **Apply**.

H.323 to SIP Calls

To configure a Q.850 reason with cause to SIP status mapping:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click SIP-Q850 Map.
3. Click **Add** to add the configuration. The SIP > H.323 calls table appears in the content area.
4. Click **Add**. The Add SIP Q.850 Map Entry dialog box appears.
5. **Q850 cause code**—Enter the Q.850 cause code that you want to map to a SIP status and reason. The valid range of values is 100-699.

6. **SIP status code**—Enter the SIP status code to which you want to map the Q.850 cause code. The valid range of values is 100-699.
7. **SIP reason**—Enter the SIP reason that you want to use with the SIP status code. If your values have spaces between characters, enclose your entry within quotation marks.

Dialog box titled "Add session-router->q850-sip-map->entry".

*Q850 cause code	31	(Unique required. Range:0..999999. Default:0)
*SIP status code	250	(Range:100..699,..0..0. Default:0)
*SIP reason	Temporarily Unavailable	

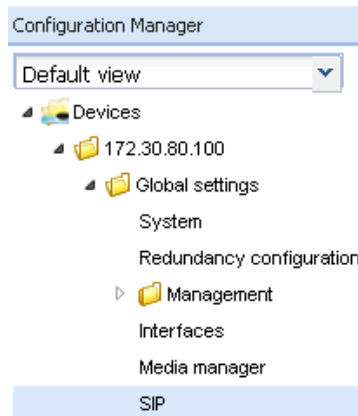
Buttons: Apply, Cancel

8. Click **Apply**. The new entry appears in the H.323->SIP calls table.
9. Click **Apply**.

Enabling the Net-Net SBC to add the Reason Header

To enable the Net-Net SBC to add the Reason header to IWF calls:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **Reason headers in SIP responses**—Select enabled from the combination box.

Reason headers in SIP responses enabled

5. Click **Apply**.

Trunk Group URIs

The Net-Net SBC can enable the trunk group capabilities described in this section (applicable for SIP and IWF signaling services). This implementation follows the IPTEL draft “Representing Trunk Groups in Tel/SIP Uniform Resource Identifiers (URIs)” (draft-ietf-ipitel-trunk-group-06.txt), and also supports more customized approaches.

- For a typical access call flow scenario, when the calling party’s call arrives at the Net-Net SBC, the Net-Net SBC formulates a SIP INVITE message that it sends to a softswitch. The Net-Net SBC supports a URI contact parameter in the SIP request message so that service providers need to be able to:
 - Determine from where the Net-Net SBC received the call
 - Signal information about the originating gateway from a Net-Net SBC to a softswitch (for example, an incoming trunk group or a SIP gateway to a Net-Net SBC)
- Support of signaling of routing information to the Net-Net SBC from network routing elements like softswitches. This information tells the Net-Net SBC what egress route (or outgoing trunk groups) it should choose for terminating next hops/gateways. For this purpose, SIP URI parameters in the Request-URI are defined. Additional URI parameters include the network context to identify the network in which the originating or terminating gateway resides.
- Net-Net SBC resources are freed by reducing the number of local policy, session agent, and session agent group configurations (especially important for large business applications). By enabling the trunk group URI feature, the Net-Net instead uses a routing scheme based on signaled SIP URI information.

Terminology

The following IPTEL terms are used in the descriptions of and instructions for how to configure this feature:

- **Trunk**—In a network, a communication path connecting two switching systems used in the establishment of an end-to-end connection; in selected applications, it may have both its terminations in the same switching system
- **Trunk group**—A set of trunks, traffic engineered as a unit, for the establishment of connections within or between switching systems in which all of the paths are interchangeable except where sub-grouped
- **Trunk group name**—Provides a unique identifier of the trunk group; referred to as `trgp`
- **Trunk group context**—Imposes a namespace by specifying a domain where the trunk groups are; also referred to simply as “context”

Trunk Group URI Parameters

Trunk group URI parameters identify originating and terminating trunk group information in SIP requests.

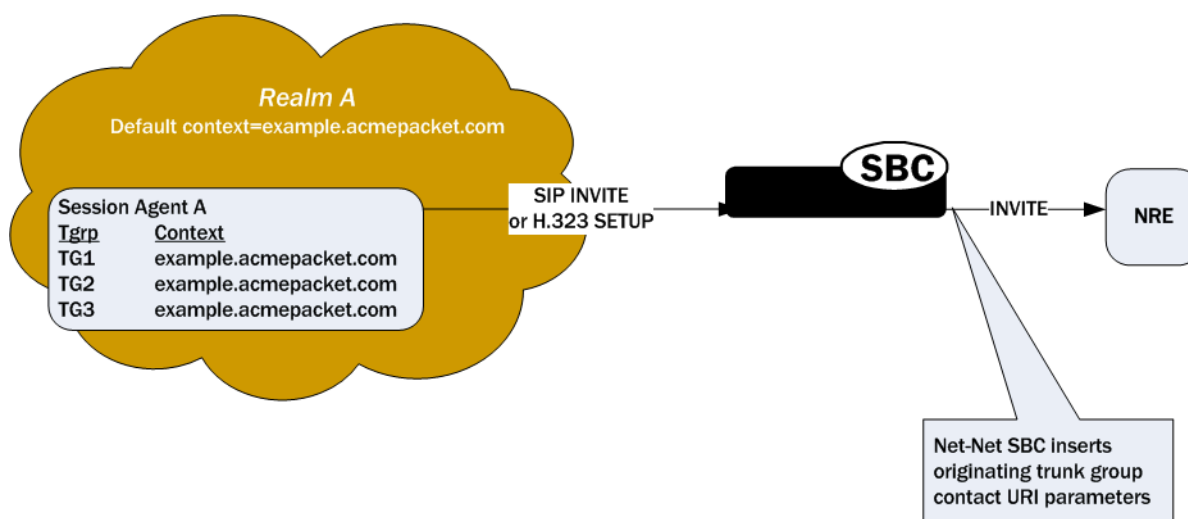
In the absence of official SIP standards for transporting trunk groups between signaling elements, the Net-Net SBC allows you to define URI parameters for use with originating and terminating trunk group URIs.

Originating Trunk Group URI Parameters and Formats

You can configure session agents and session agents groups on the Net-Net SBC to insert trunk group URI parameters in the SIP contact header. When SIP gateways comply with the IPTEL draft, they include the originating URI parameter in the SIP contact header. For those SIP and H.323 gateways that are not compliant, the Net-Net SBC inserts SIP trunk group URI parameters on the gateway's behalf.

When there are no applicable session agent or session agent group configurations, the Net-Net SBC uses the source IP address of the endpoint or gateway as the trunk group name (trgp) parameter in the originating trunk group URI.

The following diagram shows a scenario where the Net-Net SBC inserts originating trunk group URI parameters.



There are two available formats for the originating trunk group URIs:

1. In compliance with the IPTEL draft, the first format has two parameters: trgp (identifier of the specific trunk group) and trunk-context (defines the network domain of the trunk group). These appear in the following formats:

- trgp="trunk group name"
- trunk-context="network domain"

The URI BNF for would appear as it does in the example directly below, where the trgp is tg55 and the trunk-context is trunk-context is telco.example.com:

tel:+15555551212;trgp=tg55;trunk-context=telco.example.com

2. The second format is customized specifically for access URIs and contains two provisioned parameters: trgp (or tgname) and context (or provstring). This appears as trgp.context, where these definitions apply:

- trgp (tgname)—Provisioned trunk group name for the originating session agent; this value must have at least one alphabetical character, cannot contain a period (.), and can contain a hyphen (-) but not as the first or the last character
- context (provstring)—Name of the originating trunk group context; this value must have at least one alphabetical character in the top label

This format conforms to format for a hostname in the SIP URI as specified in RFC 3261, such that a trunk group identifier would appear as:

```
custsite2NY-00020.type2.voip.carrier.net
```

where the trgp is custsite2NY-00020, and the context is type2.voip.carrier.net.

The BNF for an access URI conforms to the following:

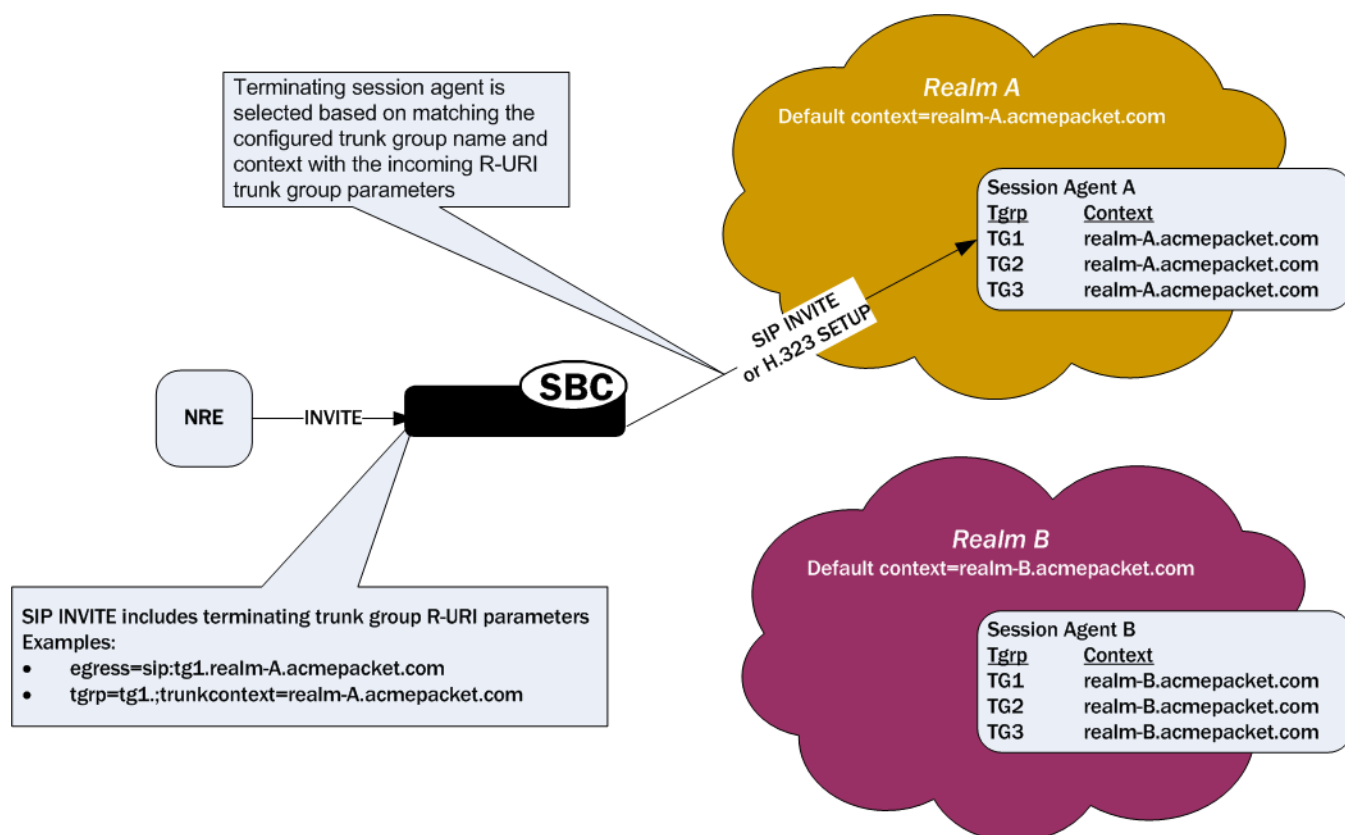
```
SIP-URI = "sip:" [userinfo ] hostport uri-parameters [headers ]
uri-parameters = *( ";" uri-parameter )
uri-parameter = transport-param / user-param / method-param
/ ttl-param / maddr-param / lr-param / other-param

other-param = accessid / pname [ '=' pvalue ]
accessid = "access=" accessURI

accessURI = scheme tname [ "." provstring ]
scheme = "sip:" / token
tname = ALPHA / *(alphanum) ALPHA *(alphanum / "-") alphanum /
alphanum *(alphanum / "-") ALPHA *(alphanum) # up to 23 characters
provstring = *(domain ".") toplabel # up to 24 characters
toplabel = ALPHA / ALPHA *( alphanum / "-") alphanum
domain = alphanum/ alphanum *( alphanum / "-") alphanum
```

Terminating Trunk Group URI Parameters and Formats

Terminating trunk group URI parameters appear in the R-URI, and they can be included in by a network routing element to instruct the Net-Net SBC which egress trunk groups to use. By matching the trunk group URI parameter with configured session agents or session agent groups, the Net-Net SBC can locate the terminating gateway. The trunk group name can also be expressed as the IP address of the terminating gateway.



In the absence of official SIP standards for transporting trunk groups between signaling elements, the Net-Net allows you to define the URI parameters used in terminating trunk groups.

There are two available formats for the terminating trunk group URIs:

1. In compliance with the IPTEL draft, the first format has two parameters: tgrp (which can be either a trunk group name or an IP address) and trunk-context (defines the network domain of the trunk group). These appear in the following formats:

- tgrp="trunk group name"
- trunk-context="network domain"

An example R-URI with terminating trunk group parameters appears as follows, where the tgrp is TG2-1 and the context is isp.example.net@egwy.isp.example.net:

```
INVITE sip:+15555551212;tgrp=TG2-1;trunk-context=isp.example.net@egwy.isp.example.net SIP/2.0
```

2. The second format is customized specifically for egress URIs and contains two provisioned parameters: tgrp (or tgname) and context (or tgdomain). This appears as tgrp.context (or tgname.tgdomain), where definitions apply:
 - tgrp (tgname)—Provisioned trunk group name for the originating session agent; this value must have at least one alphabetical character, cannot contain a period (.), and can contain a hyphen (-) but not as the first or the last character

- context (tgdomain)—Name of the terminating trunk group context; this value can be up to twenty-four characters

The use of multiple terminating trunk groups is not supported.

The BNF for a single, egress URI with trunk group information conforms to:

```

SIP-URI = "sip:" [userinfo ] hostport uri-parameters [headers ]
uri-parameters = *( ";" uri-parameter )
uri-parameter = transport-param / user-param / method-param
/ ttl-param / maddr-param / lr-param / other-param

other-param = egressid / pname [ '=' pvalue ]
egressid = "egress=" egressURI
egressURI = scheme tname [ "." tgdomain ]
scheme = "sip:" / token
tname = ALPHA / *(alphanum) ALPHA *(alphanum / "-") alphanum /
alphanum *(alphanum / "-") ALPHA *(alphanum) # up to 23 characters
tgdomain = *(domain ".") toplevel # up to 24 characters
toplevel = ALPHA / ALPHA *( alphanum / "-") alphanum
domain = alphanum/ alphanum *( alphanum / "-") alphanum

```

How It Works

For all trunk group URI support, you must set the appropriate parameters in the SIP manipulations configuration and in the session agent or session agent group configurations.

In the originating trunk group URI scenario, a call arrives at the Net-Net SBC from a configured session agent or session agent group. If this session agent or session agent group has the appropriate trunk group URI parameters and inbound manipulation rules configured, the Net-Net SBC then looks to the SIP manipulations configuration and add the trunk group URI information according to those rules. Those rules tell the Net-Net SBC where and how to insert the trunk group URI information, and the Net-Net SBC forwards the call.

In the terminating trunk group scenario, a call arrives at the Net-Net SBC from, for instance, a call agent. This call contains information about what trunk group to use. If the information matches a session agent or session agent group that has outbound manipulation rules configured, the Net-Net SBC will then look up the SIP manipulations configuration and strip information according to those rules. Those rules tell the Net-Net SBC where and how to remove the information, and the Net-Net SBC forwards the call.

SIP Header and Parameter Manipulation

SIP header and parameter manipulation is its own configuration where you can set up rules for the addition, removal, and modification of a SIP header or the elements of a SIP header. For example, you can set up the configuration to add a URI parameter to the URI in a SIP header or replace an FQDN with an IP address. For trunk group URI support, this configuration tells the Net-Net SBC where and how to manipulate the SIP message to use originating (access) and terminating (egress) trunk group URI parameters.

These manipulations can be applied at the realm or at the session agent level.

Trunk Group Routing

You can configure SIP interfaces (using Route based on trunk group parameters) to perform routing based on the trunk group information received in SIP requests. There are three options: none, IPTEL, and egress URI.

- If you leave Route based on trunk group parameters set to none (its default), the Net-Net SBC will not look for or route based on terminating trunk group URI parameters
- When you set Route based on trunk group parameters to either **iptel** or **egress-uri** and the incoming request has the trunk group parameter of this type (IPTEL or egress URI), the Net-Net SBC will select the egress next hop by matching the “tgrp” and “trunk context” with a configured session agent or session agent group.

If the received terminating trunk group URI parameters include an IP address, the egress next hop is the IP address specified. The Net-Net SBC determines the egress realm by matching the trunk context it receives with the trunk context you configure for the realm.

- If the incoming request does not have trunk group parameters or it does not have trunk group parameters of the type that you configure, the Net-Net SBC uses provisioned procedures and/or local policy for egress call routing.

The Net-Net SBC returns errors in these cases:

- If the terminating trunk group URI parameters do not identify a local Net-Net SBC session agent or session agent group, then the Net-Net SBC returns a SIP final response of “488 Not Acceptable Here.”
- If the Net-Net SBC receives a SIP INVITE with terminating trunk group URI parameters that do not match the specified syntax, the Net-Net SBC returns a 400 final response with the reason phrase Bad Egress=Parameters.

Trunk Group URIs and SIP Registration Caching

For calls where SIP registration caching is used, you will need to set certain parameters that enable the Net-Net SBC to preserve trunk group URI parameters on the outgoing side.

- For SIP-SIP calls, you set the **preserve-user-info option** in the SIP interface configuration.
- For SIP-H.323 calls requiring IWF, you set the **preserve-user-info-sa option** in the session agent configuration.

Before You Configure

Before you configure your Net-Net SBC to support trunk group URIs, you need to determine:

- How you want to manipulate SIP headers (entered in the SIP header manipulations configuration)
- For terminating trunk group routing, the trunk group mode you want to use (none, IPTEL, or egress URI); this decides routing based on trunk group information
- The trunk group name and context to use entered in a session agent or session agent group configuration
- Whether you are using originating or terminating trunk group URIs (entered in the session agent configuration)
- The trunk group context for use in a realm configuration, in case the trunk group name in the session agent or session agent group does not have a context

Configuring SIP Manipulations

For detailed instructions about how to configure SIP header manipulations, refer to the Header and Parameter Manipulation section of this guide.

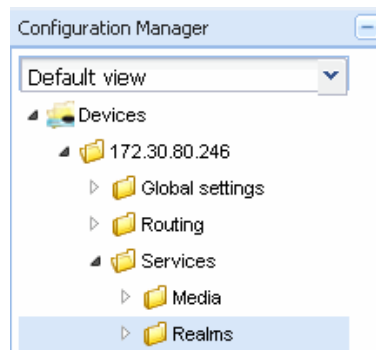
When you configure the SIP header manipulations to support trunk group URIs, take note of:

- The name of the configuration, so that you can use it when you apply the manipulations in a session agent for the inbound or outbound manipulations
- The **new-value** parameter, which specifies the trunk group and trunk group context that you want to manipulate; the possible values that apply to trunk group URI configurations are \$TRUNK_GROUP and \$TRUNK_GROUP_CONTEXT

Setting the Trunk Group URI Mode for Routing

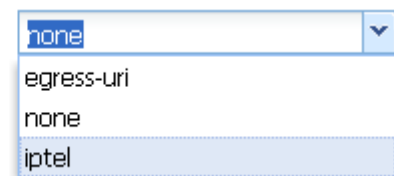
To set the trunk group URI mode for routing:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. **Route based on trunk group parameters**—Select a mode from the combination box. For example:

Route based on trunk group parameters



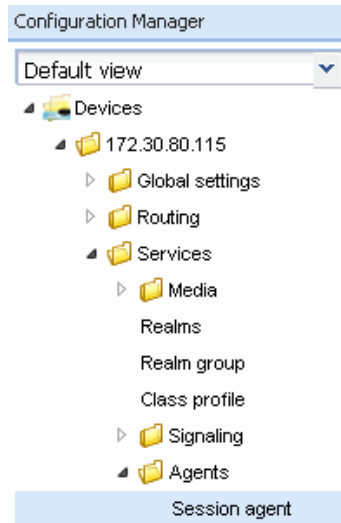
- **egress-uri**—Uses trunk group URI routing based on the egress URI format
 - **none**—Disables routing based on trunk groups
 - **iptel**—Uses trunk group URI routing based on the IPTEL formats
7. Click **Apply**.

Configuring a Session Agent for Trunk Group URIs

You can configure the outbound or inbound SIP header manipulation rules to use, as well as a list of trunk group names.

To configure a session agent for trunk group URIs:

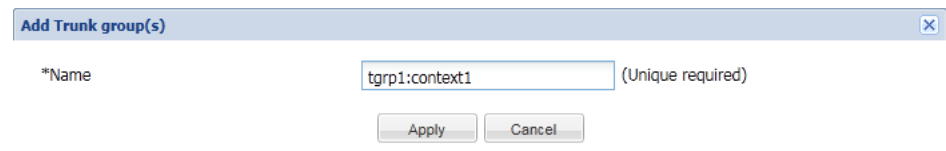
1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the session agent you want to configure in the table and click **Edit**.
4. Scroll to the Trunk group(s) table and click **Add**. The Add trunk group(s) dialog box appears.
5. **Name**—Enter the trunk group name and trunk group context in one of the following formats:

tgrp:context

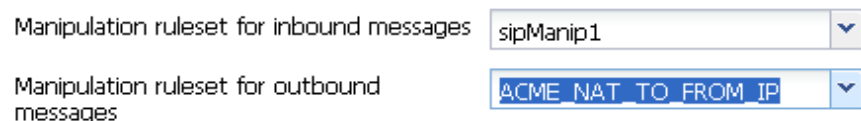
tgrp.context



Note: If you do not include a context, the Net-Net SBC uses the trunk group context you set in the realm.

6. Click **Apply**. The new entry appears in the Trunk group(s) list.
7. Click the Translation and manipulation tab at the top of the content area.
8. **Manipulation ruleset for inbound messages**—Select the SIP header manipulation you want to apply to the traffic entering the Net-Net SBC through this session agent from the combination box.
9. **Manipulation ruleset for outbound messages**—Select the SIP header manipulation you want to apply to the traffic exiting the Net-Net SBC through this session agent from the combination box.

Manipulation ruleset

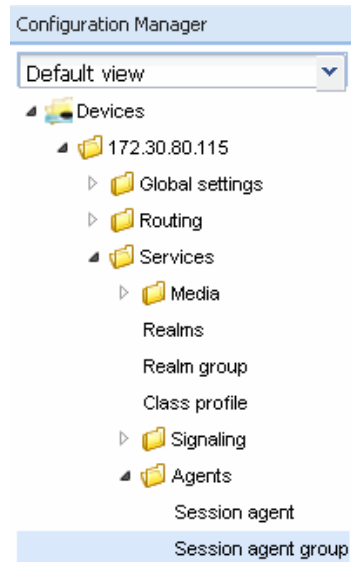


- Click **Apply**.

Configuring a Session Agent Group for Trunk Group URIs

To configure a session agent group for trunk group URIs:

- Expand the Configuration Manager slider > Services > Agents.



- Click Session agent group. The Session agent group table appears in the content area.
- Select the session agent group whose trunk group URI list you want to configure and click **Edit**.
- Scroll to the Trunk group(s) table and click **Add**. The Add trunk group(s) dialog box appears.
- Name**—Enter the trunk group name and trunk group context in one of the following formats:

tgrp:context
tgrp.context

Note: If you do not include a context, the Net-Net SBC uses the trunk group context you set in the realm.

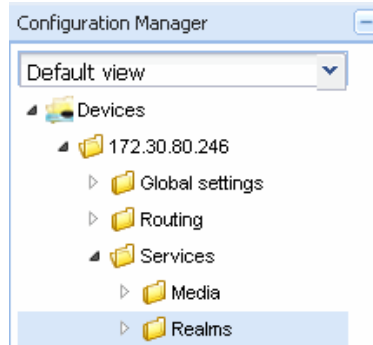
- Click **Apply**. The new entry appears in the Trunk group(s) list.
- Click **Apply**.

Setting a Trunk Group Context in a Realm

You can set trunk group contexts at the realm level, which will be used by all session agents and session agent groups if there is no context specified in their configurations.

To set a trunk group context in a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. **Trunk context**—Enter the context you want to set.

Trunk context

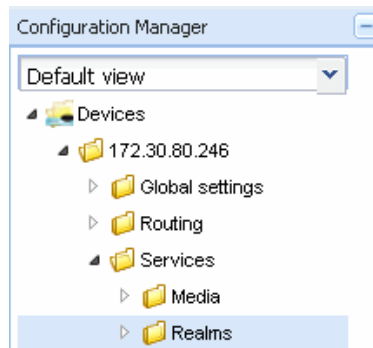
5. Click **Apply**.

Trunk Group URIs with SIP Interface Registration Caching

If you are using trunk group URIs with SIP interfaces that have registration caching enabled, then you need to configure the `preserve-user-info` option for that SIP interface.

To configure `preserve-user-info`:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll down to the Options section and click **Options**.

7. Scroll to User defined options and type **preserve-user-info** in the textbox, making sure that the syntax is correct.

User defined option	preserve-user-info	Add

8. Click **Add** to add the option. The preserve-user-info-sa window appears in the middle of the SIP options window.
9. Click **OK**.
10. Click **Apply**.

Emergency Session Handling

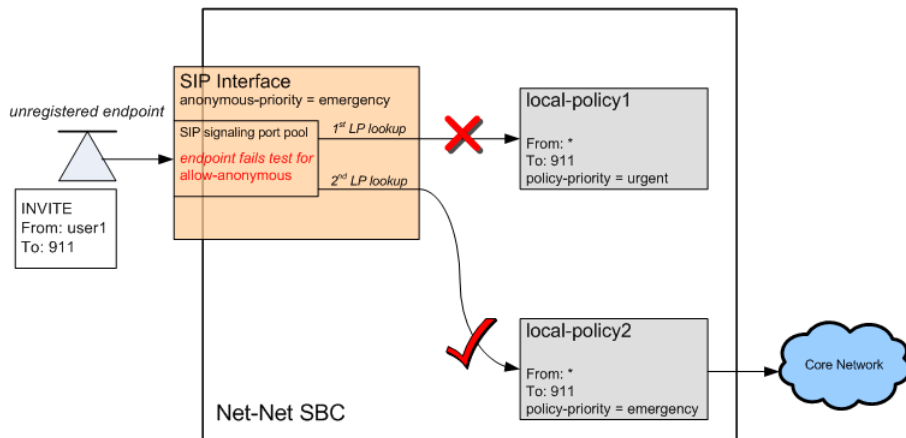
The Net-Net SBC provides a mechanism to handle emergency sessions from non-allowed endpoints. An endpoint is designated as non-allowed if it fails the admission control criteria specified by the allow-anonymous parameter value for SIP ports.

When the Net-Net SBC receives a non-allowed emergency request, it performs a local policy lookup for a matching local policy. An emergency local policy could be configured to match if the To: header in a SIP message was addressed to 911.

Both the SIP interface and the local policy have emergency policy priority selection criteria. In the SIP interface, the parameter is called anonymous-priority. In the local policy, the parameter is called policy-priority.

For the Net-Net SBC to choose a local policy to route an emergency call, the emergency policy priority value on the local policy must be equal to or greater than the emergency policy priority value on the SIP interface where the emergency message was received. In this scheme, an emergency policy priority value of none is the lowest value and an emergency policy priority value of emergency is the highest.

When a match is made between all existing local policy criteria and the emergency policy priority, the emergency call will be sent to the core network according to the chosen local policy. In addition, the policy priority value of the chosen local policy is inserted into the Priority header of the core-bound SIP message.



Emergency Session Handling Configuration Procedures

Note the value of the allow anonymous parameter in the SIP interface's SIP ports for the incoming interface you are configuring. When an incoming emergency call from an unregistered endpoint can not be characterized by this setting, the Net-Net SBC will use the following means to route the call.

1. Set the anonymous priority for the incoming SIP interface. This priority specifies that for an INVITE received from an anonymous endpoint, the Net-Net SBC will choose a local policy of equal or greater policy priority for outbound routing.
2. Set the policy priority for the local policy. Most likely, this local policy will route messages to SIP devices that act on emergency calls. The local policy is selected when its value (or above) matches the anonymous priority value for the SIP interface that receives the incoming phone call from an unregistered endpoint.

The enumerated values for both the anonymous priority and policy priority are: none, normal, non-urgent, urgent, emergency.

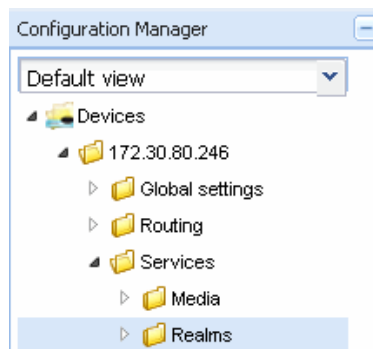
Configuring Emergency Session Handling

You can configure emergency session handling using Net-Net Central.

Setting the Anonymous Priority for the SIP Interface

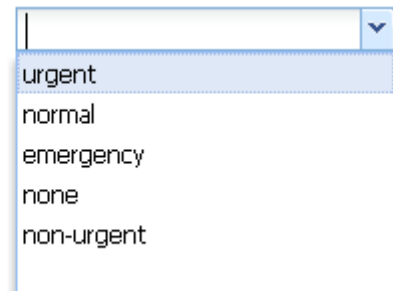
To set the anonymous priority:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. **Minimum priority for routing messages from anonymous**—Select the priority you want to set on messages from anonymous endpoints from the combination box.

Minimum priority for routing messages from anonymous

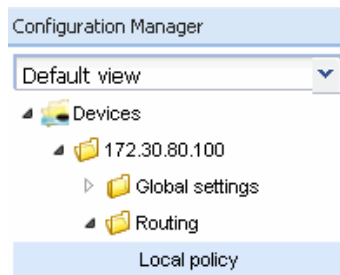


7. Click **Apply**.

Setting the Policy Priority for the Local Policy

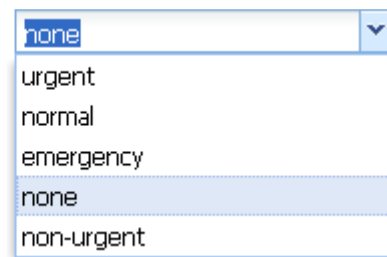
To set the policy priority for the local policy:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Click the route in the table that you want to configure and click **Edit**.
4. **Priority**—Select the priority you want for the local policy from the combination box.

Priority



5. Click **Apply**.

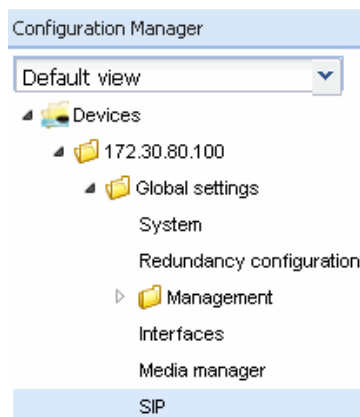
Fraud Prevention

The Net-Net SBC can constrain outgoing SIP messages to a maximum size in bytes in order to support fraud prevention techniques. If a message does exceed the configured size, it is dropped. A SIP message can be constrained from 0 to 65535 bytes, with a default value of 4096 bytes.

Configuring Fraud Prevention

To set a maximum SIP message size:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **Maximum SIP message length (bytes)**—Enter the maximum length of outgoing SIP messages in the Maximum SIP message length (bytes) textbox.

*Maximum SIP message length(bytes)

5. Click **Apply**.

Early Media Suppression

This section explains how to configure SIP early media suppression, which lets you determine who can send early media and in what direction. Early media are the RTP/RTCP packets sent from the called party to the caller, or vice versa, before a session is fully established (before a 200 OK is received). When the Net-Net SBC receives an INVITE message with SDP, it can forward media packets to the calling endpoint as soon as it forwards the INVITE to the next hop. It can also forward media packets received from the calling endpoint to the called endpoint as soon as the Net-Net SBC receives SDP in a SIP response to the INVITE, usually a provisional message. This allows for any early media to be played, such as remote ringback or announcement.

Early media can be unidirectional or bidirectional, and can be generated by the caller, the callee, or both.

With early media suppression, you can block early media until the call is established. You can define which outbound realms or next hop session agents are allowed to send or receive early media. Early media suppression only applies to RTP packets. RTCP packets received by Net-Net SBC are still forwarded to their destination in both directions, unless an endpoint is behind a NAT and the media manager has not been enabled for RTCP forwarding.

Note: To use early media suppression, you cannot configure media release of any kind: same-realm, same-network, or multiple-system media release.

How it Works

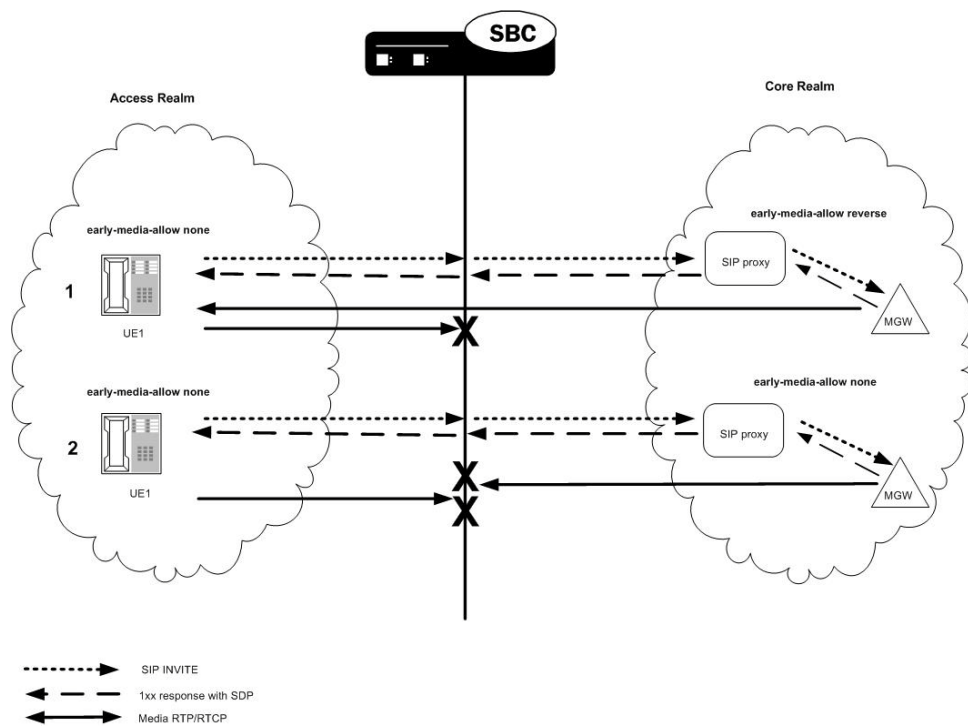
With the SIP-based addressing, early media suppression is based on the outbound SIP interface realms and the value of their early-media-allow parameter. When the Net-Net SBC forwards a SIP Invite out a SIP interface, the outbound realm is chosen based on the SIP layer information, such as the session agent for the next-hop or the address prefix of the next-hop SIP device. The matching realm's early-media-allow parameter value then applies to either allow all, block all, or block one-way early media until a 200 OK is received. At that point bidirectional media is allowed. The decision is based on SIP-layer addressing of next-hops.

You configure a rule for a realm or a session agent to use early media suppression. An early media suppression rule specifies whether you want to prevent early media in any direction, allow early media going to the calling endpoint in the reverse direction, or allow early media in both directions. The forward direction is when the packets flow from the caller to the called party. The reverse direction is when the packets flow from the called party to the caller.

The early media suppression rule is applied to a session. When the Net-Net SBC initiates a new session, it first checks whether the next hop is a session agent and if so, whether an early media suppression rule has been configured on it. If an early media suppression rule is found, the Net-Net SBC enforces it. If the next hop is not a session agent or no early media suppression rule is configured, the Net-Net SBC checks whether an early media suppression rule has been configured for the outbound realm. If it finds one, it enforces it.

Example

The following illustration shows two examples of early media suppression.



1. Caller UE1 makes a call to the PSTN media gateway (MGW). The INVITE traverses from UE1 to the Net-Net SBC through the softswitch to the MGW. The Net-Net SBC allows early media from the core to reach UE1.

2. The PSTN MGW makes a call to UE1. The INVITE traverses to the Net-Net SBC and to UE1. The Net-Net SBC blocks all early media to and from UE1 until a 200 OK is received.

Early Media Suppression Support

The Net-Net SBC supports suppressing early media in the following directions no matter which side makes the SDP offer, until it receives 200 OK for an INVITE:

- Forward direction based on the outbound realm or next-hop session agent
- Forward and reverse directions based on the outbound realm or next-hop session agent

The Net-Net SBC allows all media when a 200 OK response is received for the INVITE, regardless of whether the 200 OK response contains SDP.

Call Signaling

The Net-Net SBC media manager performs early media suppression according to an early media suppression rule. No change has been made to call signaling. For SIP, the Net-Net SBC still forwards SDP received in an INVITE request or response after performing a NAT to the media connection address. After which, the Net-Net SBC is ready to receive media packets from the endpoints. If an early media suppression rule has been configured, the Net-Net SBC drops the packets going in the direction being specified by the rule.

For a H.323 to SIP call, early media suppression rule does not change how the Net-Net SBC performs H.225/Q.931 call signaling and starts the H.245 procedure (if required) to establish logical channels for early media on the H.323 leg of the call.

Suppression Duration

When early media suppression is enabled in a session, the block lasts until the session is established. For a SIP to SIP call or an H.323 to SIP call, a session is established when the Net-Net SBC receives a 200 OK response to the INVITE. A 200 OK response to the INVITE terminates early media suppression, even when it does not contain a SDP. (A 200 OK response to a PRACK or an UPDATE request does not terminate early media suppression.) After a session is established, the Net-Net SBC can receive a change in media session (for example, a re-INVITE with a new SDP) without an early media suppression rule blocking the media.

About the Early Media Suppression Rule

An early media suppression rule is configured in the form of a permission. It specifies whether early media is allowed in both directions, the reverse direction only or not at all. Reverse direction media is media sent in the upstream direction towards the calling endpoint.

Session Agent Rule

The next-hop session agent's early media suppression rule is applied regardless of whether the media packet's source or destination address is the same as the session agent's address. For example, if the session's next hop session agent is 10.10.10.5 but the SDP in a 183 response specifies 10.10.10.6 as its connection address.

Rule Resolution

When the call's next hop is a session agent and both the outbound realm of the call and the session agent have an early media suppression rule, the session agent's early media suppression rule takes precedence. If the session agent's early media suppression rule has not been configured, the outbound realm's early media suppression rule is used, if configured.

Configuring Early Media Suppression

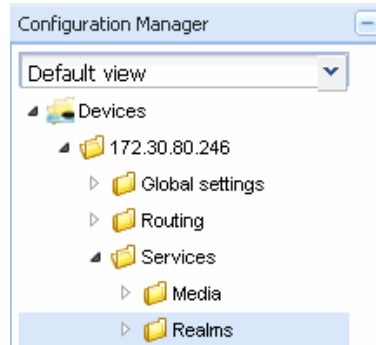
You can configure early media suppression for:

- realms
- session agents

Realm

To configure the realm:

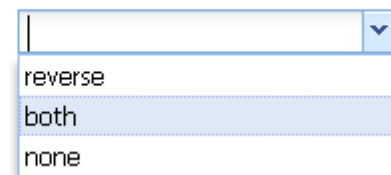
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm and click **Edit**.
4. Scroll to Early media blocking.
5. **Early media allow**—Select the early media suppression rule for the realm from the combination box:
 - **none**—No early media is allowed in either direction
 - **both**—Early media is allowed in both directions
 - **reverse**—Early media received by Net-Net SBC in the reverse direction is allowed

Early media blocking

Early media allow



There is no default value. If you leave this parameter blank, early media is allowed in either direction.

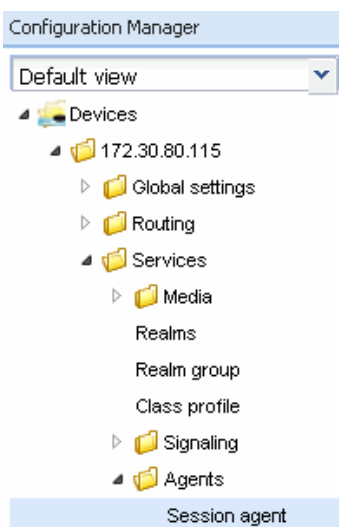
6. Click **Apply**.

Session Agents

If you do not configure early media suppression for a session agent, the early media suppression for the outbound realm is used, if configured.

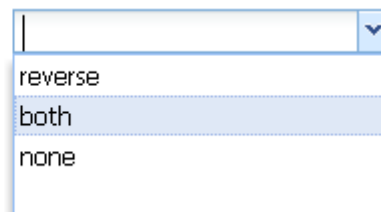
To configure session agents:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the session agent you want to configure in the table and click **Edit**.
4. Scroll to the Early media allow textbox.
5. **Early media allow**—Select the early media suppression rule for the realm from the combination box:
 - **none**—No early media is allowed in either direction
 - **both**—Early media is allowed in both directions
 - **reverse**—Early media received by Net-Net SBC in the reverse direction is allowed

Early media allow



There is no default value. If you leave this parameter blank, early media is allowed in either direction.

6. Click **Apply**.

Early Media Blocking Enhancements

You can configure the Net-Net SBC to block early media based on destination realm. Calls to such realms are prohibited from sending and receiving RTP until a SIP 200 OK response is received, and you can set the direction of the blocked media.

While decisions to block early media are customarily based on SIP-layer addressing, there are cases when the Net-Net SBC can reject early media based on the SDP address in the SDP answer for a 1XX or 2XX response. By comparing the SDP

address with the realm prefix or additional prefix address, it can block early media for matching realms. For these cases, you define global or signaling realms—ones that are not tied to SIP interfaces, but which establish additional address prefixes and rules for blocking early media.

This way, the Net-Net SBC blocks all early media for SIP interface realms, but can accept it for global realms that reference media or PSTN gateways. This configuration allows early media for calls destined for the PSTN, and blocks it for user-to-user and PSTN-to-user calls.

When configuring early media blocking consider whether you need to allow early media for certain user-to-user and PSTN-to-user calls to support, for example, custom ringback tones. The configuration also addresses the fact that Net-Net SBCs can themselves lose the ability to decide whether or not early media should be blocked when confronted with hairpinned call flows, or with traffic that traverses multiple Net-Net SBCs.

How It Works

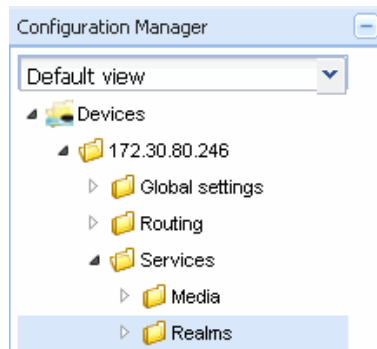
The Net-Net SBC includes a configuration called realm groups. Realm groups are sets of source and destination realms that allow early media to flow in the direction you configure. For example, you can set up realm groups to allow media from PSTN realms to user realms so that users can listen to PSTN announcements, but prohibit early media from user realms to PSTN realms.

Note that the source and destination realms you add to your lists need to be a global or signaling realm matching the caller's SDP address prefix or a SIP realm.

Configuring Realm Groups

To configure a realm group:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click **Add**. The Add Realm Group dialog box appears.

4. **Name**—Enter a name for the realm group.

5. Click **Apply** to close the dialog box. The group name appears in the realm group table.
6. Click the realm group name in the table to choose it and click **Edit**.
7. **State**—Select enabled from the combination box.

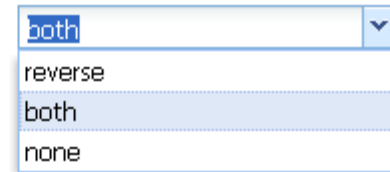
8. **Destination realms**—Click **Add**. Enter the global/SIP realm that you want to designate as destination realms for the purpose of blocking early media. This is the realm identifier value for the called SDP realms.

9. Click **OK** to close the dialog box. The destination realm name appears in the Destination realm list.
10. Repeat steps 9 through 10 to add more destination realms.
11. **Source realms**—Click **Add**. Enter the global/SIP realm that you want to designate as a source realm for the purpose of blocking early media. This is the realm identifier value for calling SDP realms.

12. Click **OK**. The source realm name appears in the Source realm list.

13. Repeat steps 12 through 13 to add source realms to the list.
14. **Direction in which to allow media**—Enter the direction for which early media is allowed for this realm group from the combination box.
 - **both** (default)—Allows early media to flow to/from called and caller
 - **reverse**—Allows early media to flow from called to caller
 - **none**—Turns off the feature for this realm group by blocking early media

Direction in which to allow early media



15. Click **Apply**.

SDP-Response Early Media Suppression

This section explains how to configure SDP-response early media suppression, which can be used when the Net-Net SBC is deployed after a softswitch or proxy in the signaling path. In this deployment, user endpoints and gateways communicate directly with the softswitch or proxy, which in turn sends call signaling to the Net-Net SBC. The call signaling gets sent back to the same or different softswitch or proxy. Because the Net-Net SBC does not communicate with the endpoints or gateways that are the media terminators, early media suppression for this deployment must use SDP-based addressing rather than the SIP-based addressing (described in the *SIP Early Media Suppression* section in this technical notice).

Using this feature lets you configure specific IP addresses for which early media should not be suppressed, based on SDP addressing. The Net-Net SBC checks the SDP addresses in SIP responses against these IP address or address ranges to determine on which media gateway a call terminates.

How it Works for SIP-Based Addressing

With the SIP-based addressing described in the *SIP Early Media Suppression* section, early media suppression is based on the outbound SIP interface realms and the value of their early-media-allow parameter. When the Net-Net SBC forwards a SIP Invite out a SIP interface, the outbound realm is chosen based on the SIP layer information, such as the session agent for the next-hop or the address prefix of the next-hop SIP device. The matching realm's early-media-allow parameter value then applies to either allow all, block all, or block one-way early media until a 200 ok is received. At that point bidirectional media is allowed. The decision is based on SIP-layer addressing of next-hops.

How it Works with SDP-Based Addressing

SDP-response early media suppression follows the same sequence described for SIP-based addressing with one exception. A provisional response with SDP media can make the Net-Net SBC select a new early-media-allow rule from another realm, based on the addressing inside the responding SDP.

When the SDP-response early media suppression feature is enabled, the Net-Net SBC searches the outbound SIP interface's realms for a matching address prefix with the connection address in the responding SDP. If it finds a match, it uses the early-

media-allow parameter value of that realm until the 200 OK message is received, then bidirectional media is allowed regardless. If the Net-Net SBC does not find a match, it searches all of the global realms for one. If it finds a match, the Net-Net SBC uses that realm's early-media-allow parameter value. If it does not find a match in the global realm(s), the Net-Net SBC continues to use the previous early-media-allow parameter value.

Global Realms

Global realms are realms that are not parents or children of any other realms, do not have defined SIP interfaces and ports (or any signaling interface or stack), and are configured to use the network interface lo0:0. They are special realms, applicable system-wide, and are currently only used for this feature. The only global realm configuration parameters applicable to early media suppression are:

- addr-prefix
- additional-prefixes
- early-media-allow
- network-interface (which must be set to lo0:0)

Additional Prefixes

You can specify additional prefixes in addition to that of the addr-prefix parameter you configure for a realm. For example, you can configure a global realm with additional address prefixes to specify the IP addresses (or ranges of addresses) of the media gateways that are allowed to send and receive early media. This overrides the SIP interface realm's early media blocking settings.

You can also enter additional prefixes in non-global realms. These additional prefixes function the same as would multiple values in the addr-prefix parameter (which only takes one value), except addresses in additional-prefixes are not used for SIP NATs.

Using the SDP-Response Early Media Suppression Rule

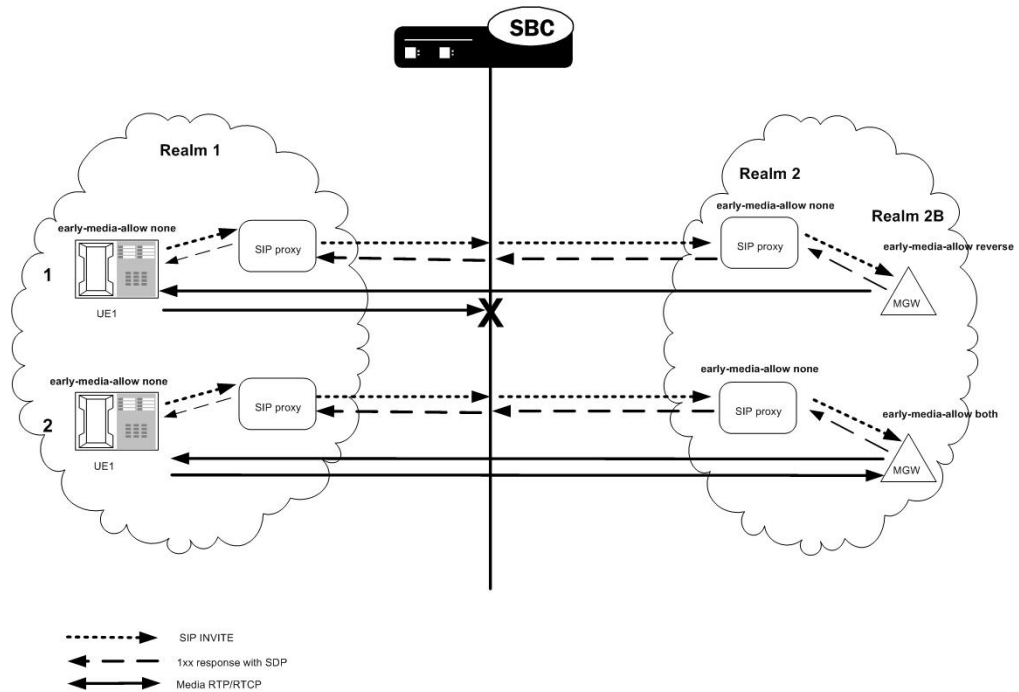
To use SDP-response early media suppression, you must add the `early-media-sdp-realms` option to the SIP interface configuration that interfaces with the next-hop device, such as the supported softswitch.

When the Net-Net SBC receives a provisional response that includes SDP from the called endpoint, and the `early-media-sdp-realms` option is active in the outgoing SIP interface of the call, it first searches the realms that apply to the outgoing SIP interface. If it does not find a realm, the Net-Net SBC searches the global realms. If the search yields a new realm that is not the SIP interface realm, its early media suppression rule (if any) replaces the existing one. Only the early media suppression rule of the new realm is applied to the call. Other realm properties from the outbound realm remain applicable to the call. If no new realm is found, the early media policy of the outgoing SIP interface realm is applied.

The Net-Net SBC allows media when the SDP media connect address in a response matches one of a configured list of IP address ranges defined in a realm and the realm has early media allowed. You need to configure specific a IP address or address range to specify which media gateways should not be suppressed based on SDP media addresses. The IP addresses are checked against the SDP being received. The decision for suppression is based on whether the matching realm allows early media. The early media will be suppressed if the matching realm does not allow early media or if there is no match and the outbound SIP interface realm does not allow early media.

Example

The following illustration shows two examples of SDP-response early media suppression.



Configuring SDP-Response Early Media Suppression

To configure SDP-response early media suppression:

1. Add the `early-media-sdp-realms` option to the SIP interface that interfaces with the softswitch.
2. Configure the SIP interface realm with an early media suppression rule that blocks all early media.
3. Configure either or both of the following:
 - One or more of the SIP realm's child realms, each with an early media suppression rule that allows `all` or `reverse` direction early media and a list of additional prefixes that specifies the IP addresses of the media gateways, or a range of IP addresses that includes the media gateways. Early media is allowed from these gateways only for calls that signals through this SIP interface.
 - One or more realms that has the network interface equal to `100:0`, an early media suppression rule that allows `all` or `reverse` direction early media and a list of additional prefixes that specifies the IP addresses of the media gateways, or a range of IP addresses that includes the media gateways. Early media is allowed from these gateways regardless of interface.

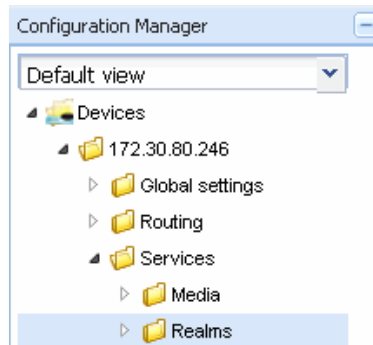
Configuring SDP-Response Early Media Suppression

You can configure SDP-response early media suppression using the Net-Net Central.

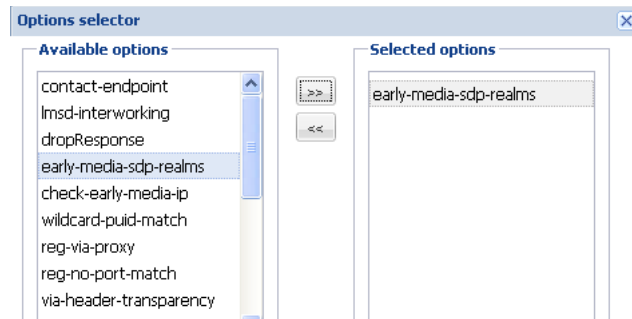
Configuring a SIP Interface

To configure a SIP interface:

1. Expand the Configuration Manager slider > Services.



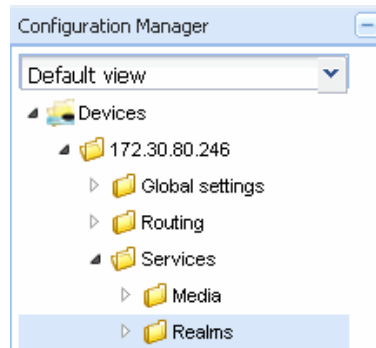
2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **early-media-sdp-realms** in the Available options list to select it.



8. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
9. Click **OK**.
10. Click **Apply**.

Configuring the Realm**To configure the realm:**

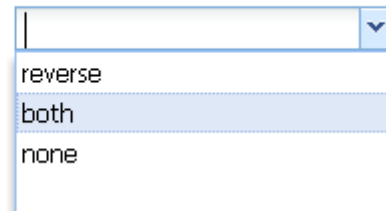
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the table and click **Edit**.
4. Scroll to the Early media blocking section.
5. **Early media allow**—Select the early media suppression rule for the realm from the combination box:
 - **none**—No early media is allowed in either direction
 - **both**—Early media is allowed in both directions
 - **reverse**—Early media received by Net-Net SBC in the reverse direction is allowed

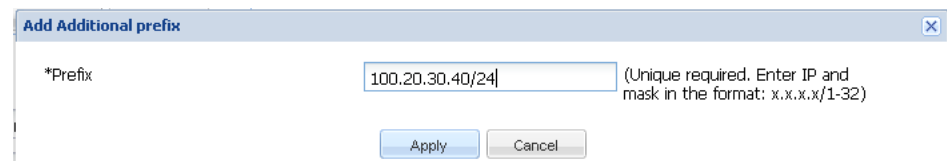
Early media blocking

Early media allow



There is no default value. If you leave this parameter blank, early media is allowed in either direction.

6. If you want to use a prefix, click **Add**. The Add Realm additional prefix dialog box appears.
7. **Additional prefix**—Enter a valid IP address and the number of bits to use to match the IP address with the address prefix.
 <IPv4> [/<number of bits>]
 Not specifying <number of bits> implies that all 32 bits are used for matching.



8. Click **Apply** to close the dialog box. The additional prefix appears in the list.
9. Click **Apply**.

SDP Anonymization

In order to provide an added measure of security, the Net-Net SBC's topology-hiding capabilities include SDP (Session Description Protocol) anonymization. Enabling this feature gives the Net-Net SBC the ability to change or modify certain values in the SDP so that malicious parties will be unable to learn information about your network topology.

To do this, the Net-Net SBC hides the product-specific information that can appear in SDP **o=** lines and **s=** lines. This information can include usernames, session names, and version fields. To resolve this issue, the Net-Net SBC makes the following changes when you enable SDP anonymization:

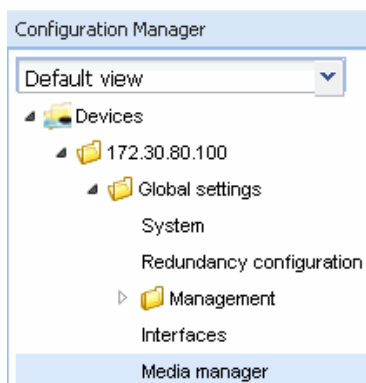
- Sets the session name (or the **s=** line in the SDP) to **s=-**
- Sets the username in the origin field to **-Net-Net SBC**
- Sets the session ID in the origin field to an integer of incrementing value

Note that for mid-call media changes, the session identifier is not incremented.

Configuring SDP Anonymization

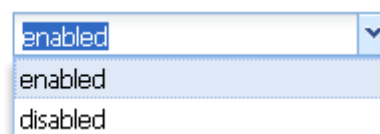
To configure SDP anonymization:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to Anonymize SDP.
5. **Anonymize SDP**—Click the checkbox to enable the SDP anonymization feature. Leaving the checkbox blank disables this feature. The default is disabled.

Anonymize SDP



6. Click **Apply**.

Restricted Media Latching

This section explains how to configure restricted media latching, which lets the Net-Net SBC latch only to media from a known source IP address, in order to learn and latch the dynamic UDP port number. The restricting IP address's origin can be either the SDP information or the SIP message's Layer 3 (L3) IP address, depending on the configuration.

About Latching

Latching is when the Net-Net SBC listens for the first RTP packet from any source address/port for the destination address/port of the Net-Net SBC. The destination address/port is allocated dynamically and sent in the SDP. After it receives a RTP packet for that allocated destination address/port, the Net-Net SBC only allows subsequent RTP packets from that same source address/port for that particular Net-Net SBC destination address/port. Latching does not imply that the latched source address/port is used for the destination of the reverse direction RTP packet flow (it does not imply the Net-Net SBC will perform symmetric RTP).

Restricted Latching

The Net-Net SBC restricts latching of RTP/RTCP media for all calls within a realm. It latches to media based on one of the following:

- SDP: the IP address and address range based on the received SDP c= connect address line in the offer and answer.
- Layer 3: the IP address and address range based on the received L3 IP address of the offer or answer. This option is for access registered HNT endpoints. If the L3 IP address is locally known and cached by the Net-Net SBC as the public SIP contact address, that information could be used instead of waiting for a response. The Net-Net SBC might use the L3 IP address restriction method for all calls regardless of whether the endpoint is behind a NAT or not, for the same realms.

Symmetric Latching

A mode where a device's source address/ports for the RTP/RTCP it sends to the Net-Net SBC that are latched, are then used for the destination of RTP/RTCP sent to the device.

How it Works

After allocating the media session in SIP, the Net-Net SBC sets the restriction mode and the restriction mask for the calling side as well as for the called side. It sets the source address and address prefix bits in the flow. It also parses and loads the source flow address into the MIBOCO messages. After receiving the calling SDP, the Net-Net SBC sets the source address (address and address prefix) in the appropriate flow (the flow going from calling side to the called side). After receiving the SDP from the called side, the Net-Net SBC sets the source address in the flow going from the called side to the calling side.

The Net-Net SBC uses either the address provided in the SDP or the layer 3 signaling address for latching. You also configure the Net-Net SBC to enable latching so that when it receives the source flow address, it sets the address and prefix in the NAT flow. When the NAT entry is installed, all the values are set correctly. In addition, sipd sends the information for both the incoming and outgoing flows. After receiving SDP from the called side sipd, the Net-Net SBC sends information for both flows to the MBCD so that the correct NAT entries are installed.

Enabling restricted latching may make the Net-Net SBC wait for a SIP/SDP response before latching, if the answerer is in a restricted latching realm. This is necessary because the Net-Net SBC does not usually know what to restrict latching to until the media endpoint is reached. The only exception could be when the endpoint's contact/IP is cached.

Relationship to Symmetric Latching

The current forced HNT symmetric latching feature lets the Net-Net SBC assume devices are behind NATs, regardless of their signaled IP/SIP/SDP layer addresses. The Net-Net SBC latches on any received RTP destined for the specific IP address/port of the Net-Net SBC for the call, and uses the latched source address/port for the reverse flow destination information.

If both restricted latching and symmetric latching are enabled, the Net-Net SBC only latches if the source matches the restriction, and the reverse flow will only go to the address/port latched to, and thus the reverse flow will only go to an address of the same restriction.

- Symmetric latching is enabled.

If symmetric latching is enabled, the Net-Net SBC sends the media in the opposite direction to the same IP and port, after it latches to the source address of the media packet.

- Symmetric latching is disabled.

If symmetric latching is disabled, the Net-Net SBC only latches the incoming source. The destination of the media in the reverse direction is controlled by the SDP address.

Example 1

A typical example is when the Net-Net SBC performs HNT and non-HNT registration access for endpoints. Possibly the SDP might not be correct, specifically if the device is behind a NAT. Therefore the Net-Net SBC needs to learn the address for which to restrict the media latching, based on the L3 IP address. If the endpoint is not behind a NAT, then the SDP could be used instead if preferred. However, one can make some assumptions that access-type cases will require registration caching, and the cached fixed contact (the public FW address) could be used instead of waiting for any SDP response.

Example 2

Another example is when a VoIP service is provided using symmetric-latching. A B2BUA/proxy sits between HNT endpoints and the Net-Net SBC, and calls do not appear to be behind NATs from the Net-Net SBC's perspective. The Net-Net SBC's primary role, other than securing softswitches and media gateways, is to provide symmetric latching so that HNT media will work from the endpoints.

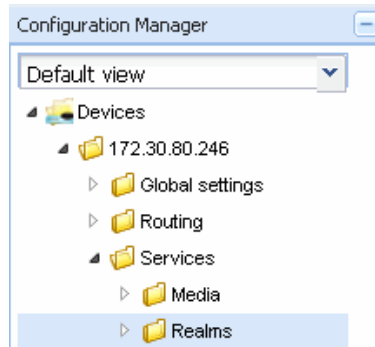
To ensure the Net-Net SBC's latching mechanism is restricted to the media from the endpoints when the SIP Via and Contact headers are the B2BUA/proxy addresses and not the endpoints', the endpoint's real (public) IP address in the SDP of the offer/answer is used. The B2BUA/proxy corrects the c= line of SDP to that of the endpoints' public FW address.

The Net-Net SBC would then restrict the latching to the address in the SDP of the offer from the access realm (for inbound calls) or the SDP answer (for outbound calls).

Configuring Restricted Latching

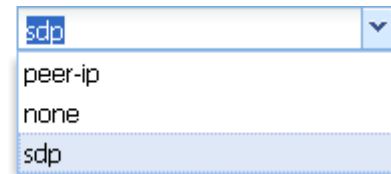
To configure restricted latching:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm from the table to which you would like to apply restricted latching.
4. Scroll to Restricted latching.
5. **Restricted latching mode**—Select the restricted latching mode:
 - **none**—No latching used
 - **sdp**—Use the address provided in the SDP for latching
 - **peer-ip**—Use the layer 3 signaling address for latching

Restricted latching mode



6. **Prefix for the source latched address**—Enter the number of address bits you want used for the source latched address. If set to 32, the complete IP address is matched. This field will be used only if restricted latching is used.

Length of bit map for source latched address

7. Click **Apply**.

Enhanced SIP Port Mapping

Enhanced SIP port mapping supports anonymous requests from endpoints and endpoints that dynamically change transport protocols between UDP and TCP.

Anonymous Requests

If a SIP endpoint sends an INVITE message with a From header that is anonymous, the Net-Net SBC can find the registration cache entry by using the Contact and Via headers. In cases such as instant messaging (IM), where there is no Contact header, the Net-Net SBC can use the Via header.

The Net-Net SBC's checks whether the **reg-via-key** option is configured for the access-side SIP interface where a REGISTER is received. If the option is enabled, the

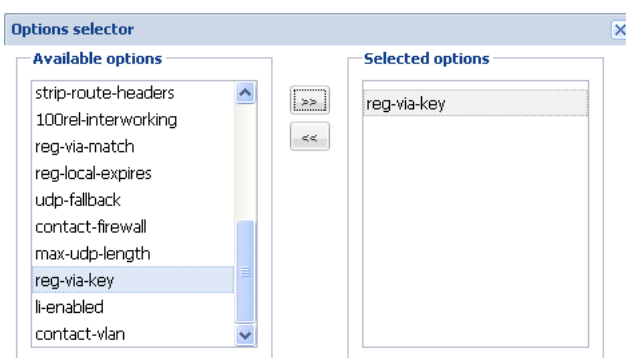
Net-Net SBC makes the via-key by adding the IP address from the Via header to the firewall address (if there is a firewall present between the Net-Net SBC and the endpoint).

When an INVITE arrives at a SIP interface where this option is enabled, the Net-Net SBC determines whether the From header is anonymous or not. If it is anonymous, then the Net-Net SBC uses the Via-key to find the registration entry.

Configuring Enhanced SIP Port Mapping

To enable support for anonymous SIP requests:

1. Click Realms in the Net-Net Central navigation tree. The Realm table appears in the display pane.
2. Click the realm to which you want to configure in the table and click **Edit**.
3. Click the Services and Protocols tab. The Realm services table appears.
4. Click the realm you want to configure in the table and click **Edit**.
5. With the SIP interface tab selected, scroll to the Options in SIP service area.
6. Click **Options**. The Options dialog box appears.
7. Click **reg-via-key** in the Available options list to select it.



8. Click  to move the option to the Selected options list.
9. The option name appears along with a description in the lower part of the window.
10. Click **OK**.
11. Click **Apply**.

Dynamic Transport Protocol Change

The Net-Net SBC uses the IP address and port in the Contact and Via headers. This is useful for cases when endpoints dynamically change transport protocols (TCP/UDP), and the port number used for sending an INVITE might not be the same one used to send a REGISTER message.

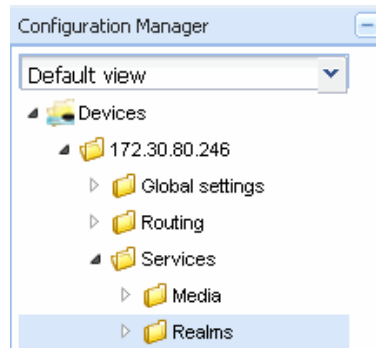
If you do not enable this feature, when an endpoint registered with the Net-Net SBC used UDP for its transport protocol, a call fails if that endpoint subsequently initiates the call using TCP. The Net-Net SBC checks for the Layer 3 IP address and port, and it rejects the call if the port is changed.

With the new option **reg-no-port-match** added to the global SIP configuration, the Net-Net SBC will not check the Layer 3 port in the INVITE and REGISTER messages.

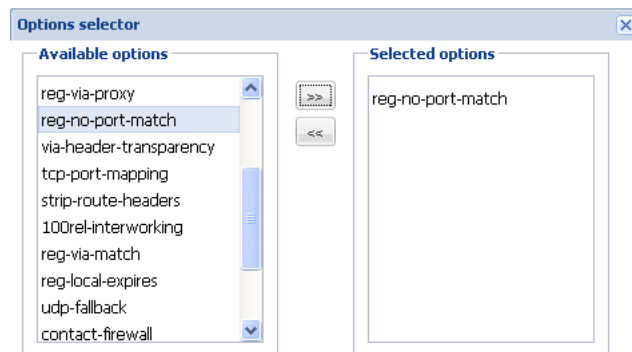
Configuring Dynamic Transport Protocol Change

To enable dynamic transport protocol change:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **reg-no-port-match** in the Available options list to select it.



8. Click  to move the option to the Selected options list.
9. The option name appears along with a description in the lower part of the window.
10. Click **OK**.
11. Click **Apply**.

Privacy Extensions

This section explains how you can configure privacy services to be applied only when the source is trusted and the destination is considered untrusted. The Net-Net SBC considers all user endpoints and nodes outside the core as untrusted.

How it Works

The Net-Net SBC acts as the boundary device between the trusted platform and the untrusted Internet, to implement privacy requirements. When it receives a message, the Net-Net SBC checks whether the source is trusted. It evaluates the level of privacy requested in a Privacy header, if present.

Depending on whether the source is trusted or untrusted, the Net-Net SBC can do different things when passing the message to the outgoing side. It also checks whether the destination is trusted.

Privacy Types Supported

The Net-Net SBC supports the following Privacy types:

- **user:** user-level privacy function provided. Any non-essential informational headers are removed, including the Subject, Call-Info, Organization, User-Agent, Reply-To, and In-Reply-To. Possibly the original value of the From header is changed to anonymous.
- **header:** headers that cannot be set arbitrarily by the user (Contact/Via) are modified. No unnecessary headers that might reveal personal information about the originator of the request are added. (The values modified must be recoverable when further messages in the dialog need to be routed to the originator.)
- **id:** third-party asserted identity kept private with respect to SIP entities outside the trust domain with which the user authenticated.

The following SIP headers can directly or indirectly reveal identity information about the originator of a message: From, Contact, Reply-To, Via, Call-Info, User-Agent, Organization, Server, Subject, Call-ID, In-Reply-To and Warning.

user

The Net-Net SBC supports the Privacy type user. It can remove non-essential information headers that reveal user information by:

- Setting the SIP From header and display information to anonymous
- Removing the Privacy header
- Removing Proxy-Require option tag = privacy (if present)
- Removing the following headers:
 - Subject
 - Call-Info
 - Organization
 - User-Agent
 - Reply-To
 - In-Reply-To

header	The Net-Net SBC also supports the Privacy type header. It modifies SIP headers that might reveal the user identity by: • Stripping the Via header • Replacing the Contact header • Stripping Record-Route • Removing the Privacy header • Removing Proxy-Require option tag = privacy (if present) In general, the B2BUA behavior of the Net-Net SBC by default provides header privacy for all sessions.
id	The Net-Net SBC also supports the Privacy type id. It keeps the Network Asserted Identity private from SIP entities outside the trusted domain by: • Stripping only P-Asserted-Identity • Removing the Privacy header and Proxy-Require option-tag = privacy • Setting the From header to anonymous (for the backward compatibility)
Examples	The following examples show the actions the Net-Net SBC performs depending on the source and target of the calls.
Calls from Untrusted Source to Trusted Target	When calls are from an untrusted source to a trusted target and PPI is included in the INVITE sent to IP border elements, the Net-Net SBC maps the PPI information to PAI in the outgoing INVITE to the trusted side (even if the Privacy header is set to id or to none). The Privacy and From headers get passed on unchanged. IP border elements must pass PAI (if received in the ingress INVITE) and the From and Privacy headers to the egress side just as they were received on the ingress side. The Net-Net SBC maps the PPI to PAI by default, if the outgoing side is trusted. To change this behavior, you need to configure the disable-ppi-to-pai option.
Calls from Trusted to Untrusted	When calls are from a trusted source to an untrusted target, and the Privacy header is set to id, the Net-Net strips PAI, makes the From header anonymous, and strips the Privacy header. If the Privacy header is set to none, the Net-Net SBC does not change the From header and passes on the Privacy header, if there is one.
Calls from Trusted to Trusted	When calls are going from trusted source to trusted target acting as a peer network border element and PPI is included, the Net-Net SBC maps PPI to PAI. The Privacy header remains the same as signaled and the Net-Net SBC passes the From header and the PAI without changes.

Configuring SIP Privacy Extensions

You configure SIP interface's trust mode as none, which means nothing is trusted for this SIP interface. You also configure the disable-ppi-to-pai parameter to disable the changing of the P-Preferred header to the P-Asserted-Identity header, if the outgoing side is trusted.

Disabling the PPI to PAI Change

To disable the changing of PPI to PAI:

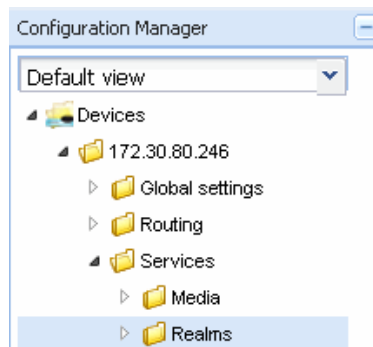
1. Change the configuration slider from Default view to ACLI view.
2. Expand the session-router folder and click **sip-config**.
3. Scroll to options and click **Options**. The Options selector dialog box appears.
4. Click **disable-ppi-to-pai** in the Available options list to select it.
5. Click  to move the option to the Selected options list. A description of this option appears in the Description field.
6. Click **OK**.
7. Click **Apply**.

Configuring SIP Privacy Extensions

You can use the Net-Net Central to configure the trust mode.

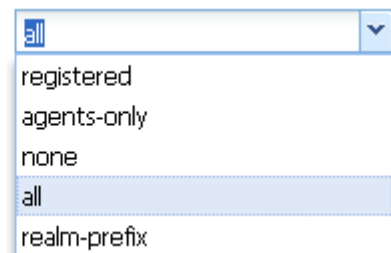
To configure the trust mode:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**. Scroll to the the Trust mode section.
6. **Trust mode**—Choose the trust mode from the combination box.

Trust mode



- **none**—Trust nothing
- **agents-only**—Trust only trusted session agents
- **registered**—Trust only trusted session agents or registered endpoints
- **realm-prefix**—Trust only trusted session agents or address matching realm prefix
- **all**—Trust all previous and next hops except untrusted session agents

7. Click **Apply**.

SIP Registration Cache Limiting

Using SIP registration cache limiting for SIP endpoint access deployments, you can restrict the size of the SIP registration cache for the global SIP configuration.

You can implement this feature if, either due to network failure scenarios or incorrect sizing of system capabilities, the Net-Net SBC and/or the SIP registrar cannot support the number of registering endpoints. Although the Net-Net SBC protects itself and the registrar against SIP REGISTER floods, conditions can still occur where too many legitimate endpoints attempt to register with the registrar via the Net-Net SBC.

By enabling SIP registration cache limiting, you restrict the number of legitimate endpoints that can register. The Net-Net SBC rejects any endpoints beyond the limit you set. If you do not want to use this feature, leave the registration cache limit parameter set to its default of 0, meaning there is no limit to the entries in the SIP registration cache.

How It Works

When you limit the number of registered endpoints allowed in the Net-Net SBC's registration cache, the Net-Net SBC analyzes each registration before starting to process it. First, the Net-Net SBC checks the contact header to determine if it is already in the list of contacts for the user. If it finds the contact in its cache list, the Net-Net SBC treats the registration as a refresh; it treats any other headers as new. Note that the Net-Net SBC checks the message prior to making any changes to the cache because it must either accept or reject the message as a whole.

The Net-Net SBC adds the number of new contacts to the number already present in the cache, and rejects any registration with a contact that would cause it to exceed its limit. Rejection causes the Net-Net SBC to send a response communicating that its registration cache is full. The default response is the **503 Registration DB-Full** message, but you can use the SIP response mapping feature to use another message if required.

You can set an option in the SIP configuration that defines the value in the Retry-After header: `reg-cache-lim-retry-after`. The Net-Net SBC sends this header as part of its rejection response when the registration cache is full. Another option sets the percentage of the registration cache size which, if exceeded, causes the Net-Net SBC to send an alarm.

About Registration Cache Additions, Modifications, and Removals

When it receives a REGISTER message with new contact information for a user, the Net-Net SBC considers it an addition to the cache and augments the number of registration cache entries. Then the Net-Net SBC forwards the message to the registrar, and—when and only when the registrar returns both the original and new

contacts in the 200 OK—the registration cache count stays the same. However, if the registrar returns only the new contact (making this a case of modification), then the Net-Net SBC removes the old contact information and subtracts accordingly from the number of registration cache entries.

Thus the Net-Net SBC does not know whether a REGISTER might result in an addition or a modification until it receives a response from the registrar. For this reason, the Net-Net SBC first assumes it is to make an addition, and then updates the registration cache and count when it has the necessary information from the registrar.

The registration cache count does not reflect removals during the rejection check because the Net-Net SBC ignores registration messages or expires headers with their expires values set to zero when it counts new entries. The fact that removals take place after additions and modifications means that messages which remove one contact while adding another might be rejected. That is, the addition might exceed the registration cache limit before any removal can take place to make room for it.

Registration Cache Alarm Threshold

A percentage of the registration cache limit, the registration cache alarm threshold is a configurable value you can set to trigger an alarm when the registration cache is reaching its limit. When exceeded, this threshold triggers the generation of an alarm and SNMP trap. When registrations fall back beneath the threshold, the Net-Net SBC clears the alarm and sends a clear trap.

This alarm is Major in severity, and its text reads as follows:

```
Number of contacts <registration count> has exceeded the registration
cache threshold <threshold %> of <registration cache limit value>.
```

SNMP Information

When the amount of registrations exceeds the configured limit, the Net-Net SBC sends out the `apSysMgmtRegCacheThresholdTrap`, which it clears with the `apSysMgmtRegCacheThresholdClearTrap` once the registration cache count has returned to a level within established limits. These appear in the `apSysRegCacheLimit`, a read-only object, as part of the `APSYSMGMT-MIB`.

The `apSysSipStatsActiveLocalContacts` object gives the current number of registration contact entries.

```
apSysRegCacheLimit OBJECT-TYPE
SYNTAX      Integer32
UNITS       "contacts"
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The maximum number of contacts allowed to be accepted into the
    registration cache. A value of 0 indicates no limit."
::= { apSysMgmtMIBGeneralObjects 14 }

apSysMgmtRegCacheThresholdTrap NOTIFICATION-TYPE
OBJECTS      { apSysSipStatsActiveLocalContacts,
               apSysRegCacheLimit
             }
STATUS      current
DESCRIPTION
    "The trap will be generated when the number of contacts stored
    in the registration cache exceeds the configured threshold."
```

```

::= { apSystemManagementMonitors 46 }

apSysMgmtRegCacheThresholdClearTrap NOTIFICATION-TYPE
    OBJECTS          { apSysSipStatsActiveLocalContacts,
                      apSysRegCacheLimit
                      }
    STATUS           current
    DESCRIPTION
        "The trap will be generated when the number of contacts stored
          in the registration cache falls below the configured
        threshold."
::= { apSystemManagementMonitors 47 }

```

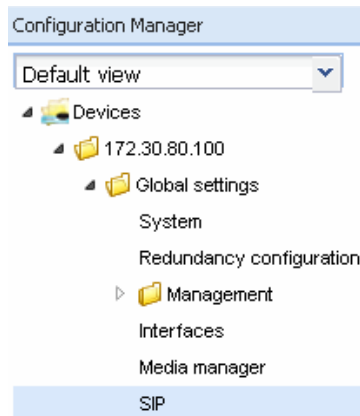
Notes on Surrogate Registration

The Net-Net SBC does not, under any circumstances, reject surrogate registrations on the basis of the registration cache limit. However, surrogate registrations generate contacts, and so they do add to the global registration count. When the surrogate registrations add to the registration count to the extent the count exceeds the limit you configure, you will have more registrations in the cache than the configured limit.

Configuring SIP Registration Cache Limiting

To configure SIP registration cache limiting:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to Registration cache limit.
5. **Registration cache limit**—Enter the maximum number of SIP registrations that you want to keep in the registration cache. The minimum and default value for this parameter is 0, and you can set it to a maximum value of 999999999.

Leaving this parameter set to 0 means there is no limit on the registration cache (and therefore leaves this feature disabled).

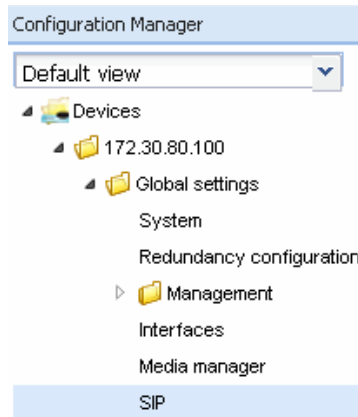
Registration cache limit

6. Click **Apply**.

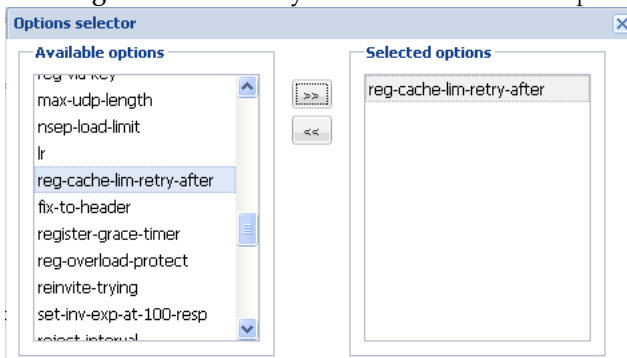
Configuring SIP Registration Cache Limiting after Retry

To configure SIP registration cache limiting after retry:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **reg-cache-lim-retry-after** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. **reg-cache-lim-retry-after**—Edit the textbox to set the registration cache limit retry after option value. This option defaults to 1800, and you can enter values from 0 to 999999999. For example:

Option

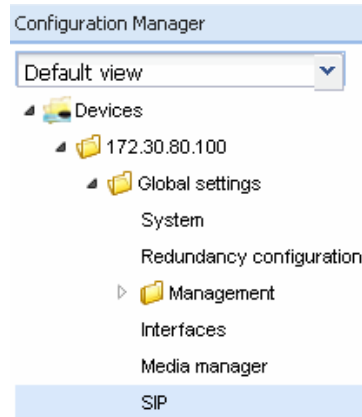
reg-cache-lim-retry-after=99999

8. Click **OK**.
9. Click **Apply**.

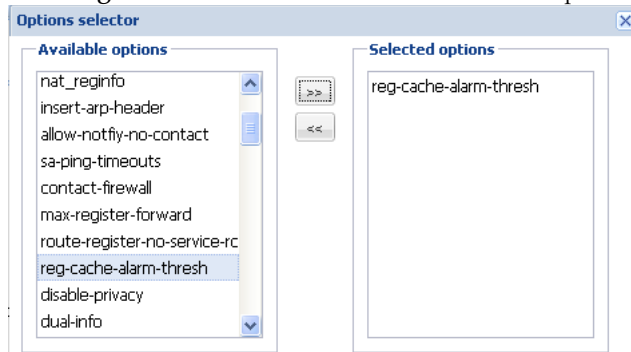
Configuring Registration Cache Alarm Threshold

To configure the SIP registration cache alarm threshold:

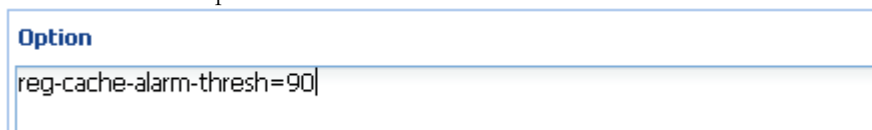
1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.
5. Click **reg-cache-alarm-thresh** in the Available options list to select it.



6. Click **>>>** to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. **reg-cache-alarm-thresh**—Edit the textbox to set the registration cache alarm threshold option value. This option defaults to 95, and you can enter values from 0 to 100. For example:



8. Click **OK**.
9. Click **Apply**.

Semi-Active IP Call Recording

The Net-Net SBC's IP call recording feature allows it to forward signaling and media packets for calls associated with a specific realm to a destination Call Recording Server (CRS).

Note: You must have a valid license with the IP Call Recording option enabled to use this feature. Without this license option feature, you will not be able to configure the Call recording server element.

How It Works

When you configure IP call recording, you must specify primary and backup realms. The network interfaces on which these realms reside source their packets through the realms in the CRS you are configuring. The Net-Net SBC considers a CRS valid only if network interfaces exist for each of the primary and backup networks. An invalid CRS entry will be ignored.

You must also specify addresses for media and signaling packets in your configuration. CRSs consist of both a signaling and media address, which are used to send the replicated packets. Each CRS object is configurable with both primary and secondary destinations. The secondary addresses are used as alternate destinations in case either of the primary CRS addresses is no longer reachable.

For a CRS to work, there **MUST** be either a primary media or primary signaling address configured. If both signaling and media packets are to be forwarded to the same address, you can leave the media address blank. However, if you provide only a media address, the Net-Net SBC forwards media and ignores signaling.

The destination addresses that you configure for media and signaling must be within the configured interface of their associated realm.

In a typical configuration, you only configure a CRS in the core realm of the network, however this is not always the case. If you configure a CRS in both the ingress and egress realms, the egress realm takes precedence.

You can enable High Availability by configuring a valid ping interval and method. When HA is configured, the Net-Net SBC creates an internal session agent with unique IP addresses for both primary and secondary CRSs. When signaling and media share the same address, the Net-Net SBC creates only one endpoint. You can have between 0-4 session agents depending on your configuration.

Any changes to a CRS are immediate; actively recorded sessions and newly initiated traffic are redirected to the newly configured destination. This is also the case when a CRS changes state in an HA mode. When a primary CRS enters an inactive state, the Net-Net SBC replicates signaling and media traffic to the secondary CRS immediately, if active. Likewise, if the realm configuration is configured to point to a new CRS object, the replicated packets are immediately sent to the new CRS. When a CRS is removed entirely from a configuration, call recording is stopped for all active sessions and no new sessions are recorded.

The Net-Net SBC replicates all call recorded packets according to the IP-in-IP RFC 2003 encapsulation standard.

There are five different states in which a CRS can be:

- **Unknown**—Set when a CRS is first created. A CRS should never be in this state under normal operation.

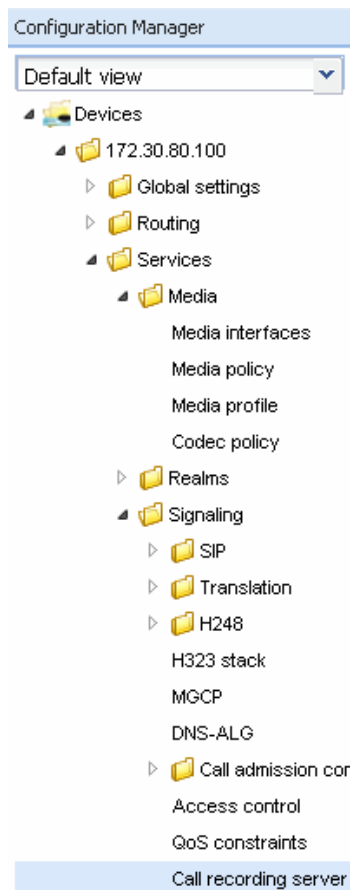
- Not Monitoring—Set when the HA mode is disabled. When in this mode, the CRS table is populated with values configured for both the primary signaling and primary media address.
- Primary Active—Set when HA mode is enabled. Always takes precedence over secondary servers if all primary session agents are in an “In Service” state.
- Secondary Active—Set when HA mode is enabled. Takes place when one or all of the primary session agents are OOS and all of the secondary session agents are “In Service.”
- No Active—Set when HA mode is enabled. Takes place when one or more of both primary and secondary session agents are OOS.

The Net-Net SBC sends SNMP traps when a change in the state of a CRS occurs. These traps include the name of the CRS and the state it is in. See the *Net-Net 4000 MIB Reference Guide* for more details.

Adding a CRS

To add a CRS:

1. Expand the Configuration Manager slider > Services > Signaling.



2. Click Call recording server.
3. Click **Add**. The Add call recording server dialog box appears.
4. **Name**—Enter the name you want to use for the CRS you are configuring in the text box.

5. **Primary media address**—Enter the primary address you want to use as a destination for forwarding media packet. If both signaling and media primary addresses the same, this parameter can be left blank
6. **Primary realm**—Choose the primary realm from the combination box from which you want this CRS to source packets from..
7. **Primary signaling address**—Enter the primary IP address you want to use as a destination for forwarding signaling packets.

Add Call recording server

*Name: callrecordserv1 (Unique required)

*Primary media address: 0.0.0.0 (Enter a valid IP address)

*Primary realm: test-realm

*Primary signaling address: 110.20.30.40 (Enter a valid IP address)

Buttons: Apply, Cancel

8. Click **Apply**.

Configuring a CRS

To configure a CRS:

1. Select the CRS you would like to configure from the table and click **Edit**.
2. Click **Edit**.
3. **Secondary realm**—Choose the backup realm from the combination box to associate with if the primary-network becomes unreachable.

Secondary realm

test-realm

OUTSIDE

4. **Secondary signaling address**—Enter the IP address you want to use as a destination for forwarding signaling packets if the address you entered in the primary signaling address parameter becomes unreachable.
5. **Secondary media address**—Enter the IP address you want to use as a destination for forwarding media packets if the address you entered in the **primary media address** parameter becomes unreachable.
6. **Ping method**—Enter the SIP method you want to use for ping messages sent to the CRS. This parameter only applies when the SIP protocol is implemented. If this parameter is left blank, HA is not used for this feature.
7. **Ping interval (secs)**—Enter the time in seconds to allow between the transmission of ping requests in an HA configuration. This parameter only applies when the SIP protocol is implemented. The default value is **0**, meaning this parameter is disabled and HA is not configured for this feature. The valid range is:
 - Minimum—0 (disabled); 2 (minimum enabled)

- Maximum—999999999

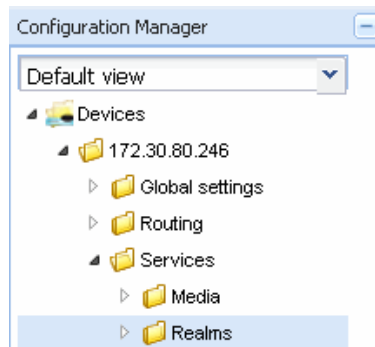
*Name	callrecordserv1
*Primary realm	test-realm ▼
*Primary signaling address	110.20.30.40
*Primary media address	0.0.0.0
Secondary realm	OUTSIDE ▼
Secondary signaling address	110.33.12.40
Secondary media address	0.0.0.0
Ping method	OPTIONS
Ping interval (sec)	0

8. Click **Apply**.

Applying a Call Recording Server Entry to a Realm

To apply a call recording server entry to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click **Realms**. The Realm table appears in the content area.
3. Click a realm in the table and click **Edit**.
4. Scroll to Call recording server.
5. **Call recording server**—Choose the name of the call recording server associated with this realm from the combination box.

Call recording server	callrecordserv1 ▼
	callrecordserv1

6. Click **Apply**.

Increased CRS Capacity

For IP call session replication recording (SRR), the Net-Net SBC can now support up to 256 call recording servers. No special configuration is required to use this increased number of CRSs.

Globally Unique Call ID

During IP call session replication recording (SRR), the Net-Net SBC records both media and signaling information and then sends it to a configured call recording server (CRS). It is the CRS's responsibility to correlate signaling messages for specific calls, which can be difficult given that the call information can traverse other network elements before reaching the CRS. The task of correlating the call information is simplified by the addition of a globally unique call ID.

For each SIP session, the Net-Net SBC can generate a unique call ID (UCID) that it inserts in SIP Request and Response messages for a call. The Net-Net SBC creates the UCID from a combination of the following put through an MD5 hash: transaction identifier, call identifier, plus the message's branch and timestamp. Each UCID is truly unique and of a fixed length.

If it receives a dialogue-initiating request (or an initial-out-of-dialog request) with the UCID already present, the Net-Net SBC uses that UCID in its X-UCID header with the name "breadcrumbs." It therefore becomes possible to trace the Net-Net SBCs traversed in that call flow if the message is recorded more than once. If breadcrumbs are present already, then the Net-Net SBC adds the current header to the end of the breadcrumbs list (as a comma-separated value) and replaces the UCID with a new one.

The ID appears as X-UCID in SIP messages, which you can see in the following examples.

UAC INVITE:

```
INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-11671-1-0
From: <sip:anonymous@anonymous.invalid>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-11671@192.168.1.60
CSeq: 1 INVITE
Max-Forwards: 70
Contact: sip:sipp@192.168.1.60:5060
X-UCID: jvo3ih1l65em8st286vdcn6sc3
Content-Type: application/sdp
Content-Length: 133
```

UAS INVITE:

```
INVITE sip:service@192.168.200.60:5060 SIP/2.0
Via: SIP/2.0/UDP
192.168.200.61:5060;branch=z9hG4bK20q0pk30581g2e06j141.1
From: <sip:anonymous@anonymous.invalid>;tag=SDe9r2601-1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: SDe9r2601-a205016a02b83cf347fefa8c0c7437a6-06a3gu0
CSeq: 1 INVITE
Max-Forwards: 69
Contact: <sip:sipp@192.168.200.61:5090;transport=udp>
```

X-UCID:
mh61vqhrukfsc9pg6sm863hjf7;breadcrumbs=jvo3ih1l65em8st286vdcn6sc3
 P-Asserted-Identity: <sip:+3901251930010@ims.vodafone.it>
 Content-Type: application/sdp
 Content-Length: 140

UAS 180 Response:

SIP/2.0 180 Ringing
 Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-11671-1-0
 From: <sip:anonymous@anonymous.invalid>;tag=1
 To: sut <sip:service@192.168.1.61:5060>;tag=SDe9r2699-2
 Call-ID: 1-11671@192.168.1.60
 CSeq: 1 INVITE
 Contact: <sip:123456@192.168.1.61:5060;transport=udp>
 Content-Length: 0
X-UCID: mh61vqhrukfsc9pg6sm863hjf7

X-UCID Notes

When you are using the globally unique call ID for SRR, remember that the Net-Net SBC:

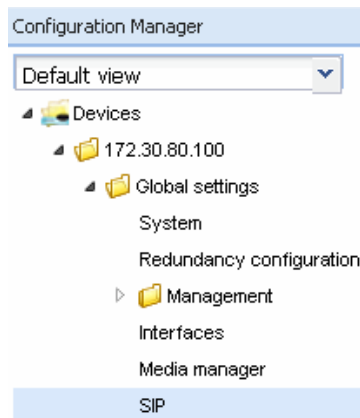
- Does not insert the X-UCID in 100 Trying messages. Since the SIP session is established after the server side processes the SIP INVITE, a 100 Trying message will have already been sent.
- Inserts or modifies the UCID after any required SIP-NAT functions are performed.
- Treats a hairpinned call as though it had traversed two Net-Net SBCs. If the UCID is added to the first INVITE, that UCID becomes a breadcrumb parameter when the Net-Net SBC hairpins the call.
- Does not insert UCIDs in out-of-dialog requests such as REGISTER or OPTIONS messages. Typically, such messages follow a single response-request model and therefore are not part of the difficulty in correlating multiple messages to a single session.

Configuring the UCID in SIP Messages

You enable and disable the use of the UCID in the global SIP configuration.

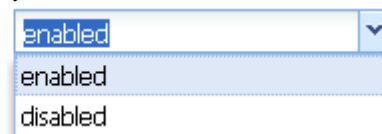
To include the UCID in SIP messages:

1. Expand the Configuration Manager slider > Global settings.



2. Click **SIP**.
3. Click **Add** to add the SIP configuration in the content area.
4. **Add unique call ID (UCID) header**—Select **enabled** from the combination box to enable this parameter if you want the Net-Net SBC to include the UCID. You can use the UCID to correlate replicated SIP message information when you use SRR. This parameter is disabled by default.

Add unique call ID (UCID) header



5. Click **Apply**.

Transport Selection

With this feature enabled, when the Net-Net SBC forwards a message larger than the value specified in the maximum UDP length parameter, it attempts to open an outgoing TCP connection to do so. This connection might fail for a number of reasons; for example, an endpoint might not support UDP, or it might be behind a firewall. The UDP fallback option addresses this condition. If it is configured in SIP interfaces associated with an outgoing message and a TCP session cannot be established, the Net-Net SBC falls back to UDP and transmits the message. When the option is not present, the Net-Net SBC's default behavior is to return the SIP status message **513 Message too Large**.

You enable this feature per SIP interface by setting options that control the maximum UDP length and allow UDP fallback:

- **max-udp-length=X** (where X is the maximum length)—Sets the largest UDP packets that the Net-Net SBC will pass. Packets exceeding this length trigger the establishment of an outgoing TCP session to deliver the packet; this margin is defined in RFC 3261. The system default for the maximum UDP packet length is 1500.

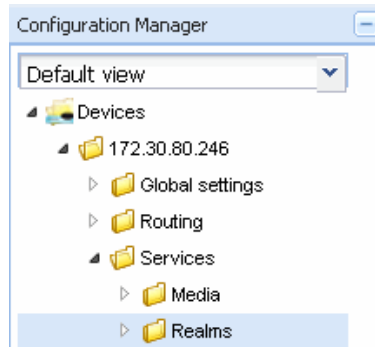
You can set the global SIP configuration's **max-udp-length=X** option for global use in your SIP configuration, or you can override it on a per-interface basis by configuring this option in a SIP interface configuration.

- **udp-fallback**—When a request needs to be sent out on the SIP interface for which you have configured this option, the Net-Net SBC first tries to send it over TCP. If the SIP endpoint does not support TCP, however, then the Net-Net SBC falls back to UDP and tries the request again.

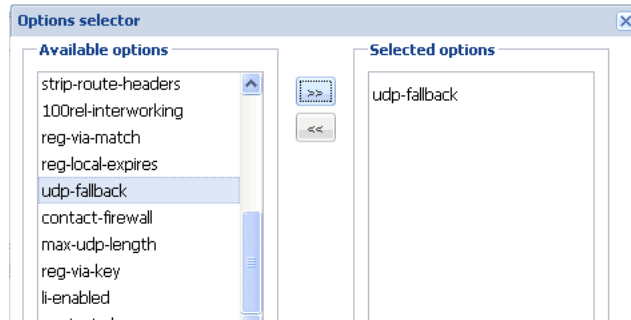
Configuring SIP Transport Selection

To configure SIP transport selection:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to the Options in SIP service section and click **Options**. The Options dialog box appears.
7. Click **udp-fallback** in the Available options list to select it.



8. Click  to move the option to the Selected options list.
9. Repeat steps 8 and 9 for the **max-udp-length** option.
10. The option names appear along with descriptions in the lower part of the window.
11. **max-udp-length**—Enter the value you want the maximum UDP length.

Option

max-udp-length=1800

Description

Syntax: max-udp-length=0. Setting max-udp-length=0 forces the sipd to send fragmented UDP packets. Using this option, you override the default value of the maximum UDP datagram size (1500 bytes; sipd requires the use of SIP/TCP at 1300 bytes).

12. Click **OK**.
13. Click **Apply**.

TCP Connection Reuse

You can configure your Net-Net SBC to reuse TCP connections created by SIP peering devices for outgoing SIP in-dialog and out-of-dialog request transactions.

The SIP draft draft-ietf-sip-connect-reuse-07.txt describes a way for SIP UAs to reuse connections created by a remote endpoint for outgoing requests for TLS. The Net-Net SBC does not support the model connection reuse is signalled by a parameter; rather, it is provisioned on a per-session-agent basis.

How It Works

You enable SIP TCP connection reuse on a per-session-agent basis. The Net-Net SBC checks incoming TCP connection request to determine if they are from session agent that has this feature turned on. When it is, the Net-Net SBC adds the connection's source address to its list of alias connections. This is a list of connections that the Net-Net SBC can use for outgoing requests rather than creating its own connection (as it does when this feature is not enabled). So if a preferred connection fails, the Net-Net SBC can refer to this list and use the alias connection.

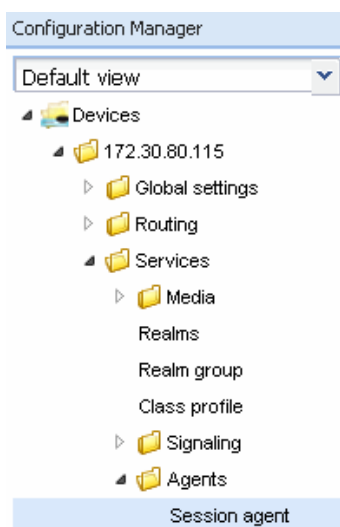
Configuring SIP TCP Connection Reuse

This section describes how to enable SIP TCP connection reuse for a session agent. Currently there are two options for the **Reuse peer TCP connection for outgoing SIP requests** parameter: **none** (which turns the feature off) and **tcp** (which enables the feature for TCP connections). You also set the reconnection interval.

To enable SIP TCP connection reuse for a session agent:

Session Agents

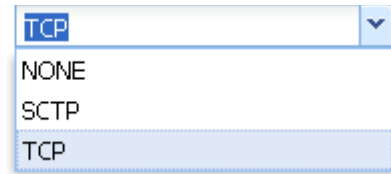
1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select and Session agent and click **Edit**.

4. **Reuse connections**—Choose TCP from the combination box to enable SIP TCP connection reuse. Leave the parameter set to **NONE** (default) if you want this feature disabled.

Reuse connections



5. **TCP reconnection interval(sec)**—Enter the amount of time in seconds to set the TCP reconnection interval.

TCP re-connection interval

180

6. Click **Apply**.

Enforcement Profile and Allowed Methods

Enforcement profile lets you configure sets of SIP methods that you want applied to: the global SIP configuration, a SIP interface, a realm, or a SIP session agent. The enforcement profile is a named list of allowed methods that you configure and then reference from the configuration where you want those methods applied.

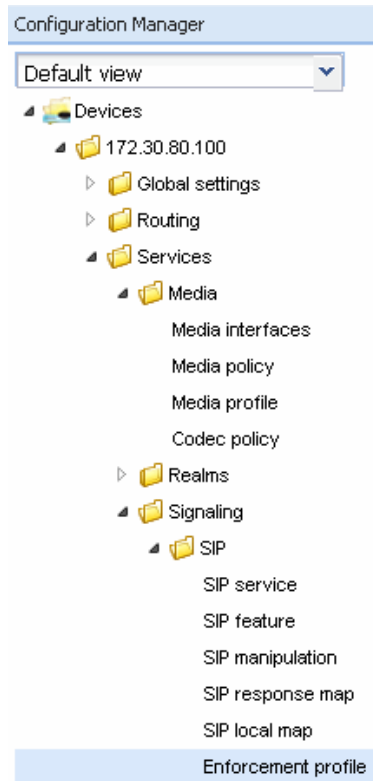
Configuring Enforcement Profiles and Allowed Methods

You need to configure an enforcement profile with a name and the list of SIP methods you want to designate as allowed. Then you need configure the global SIP configuration, a SIP interface, a realm, or SIP session agent to use the set.

Creating an Enforcement Profile

To create an enforcement profile:

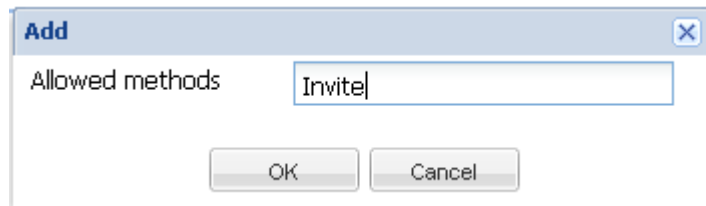
1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click Enforcement profile. The SIP enforcement profile table appears in the content area.
3. Select a profile and click **Edit**.
4. Click **Add**. The Add Enforcement Profile dialog box appears.
5. **Name**—Enter the name for the enforcement profile. You can apply this set of allowed SIP headers in: the global SIP configuration, a SIP interface, a realm, or SIP session agent.

*Name

6. Click **Add** in the Allowed methods section.
7. **Allowed methods**—Enter a SIP method that you want to allow for this profile from the combination box.



8. Click **OK** to close the dialog box. The method appears in the Allowed methods list.
9. Repeat steps 7 through 8 to add methods to the list.
10. Click **Apply**.

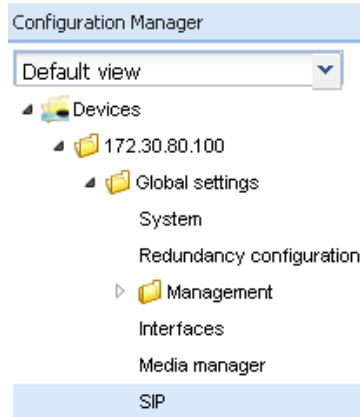
Applying SIP Enforcement Profiles

You can apply an enforcement profile to: the global SIP configuration, a SIP interface, a realm, or SIP session agent. This section shows you how to do all four. Remember that if you are adding this functionality to a pre-existing configuration, you need to select the configuration you want to edit.

Global Configuration

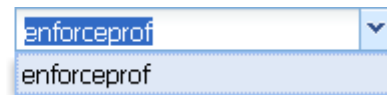
To apply an enforcement profile to the global SIP configuration:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **Enforcement profile**—Choose the name of the enforcement profile you want to apply to the global SIP configuration from the combination box.

Enforcement profile

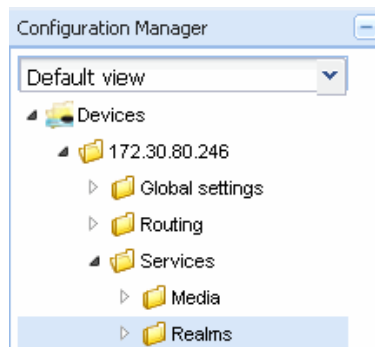


5. Click **Apply**.

SIP Interface

To apply an enforcement profile to a SIP interface:

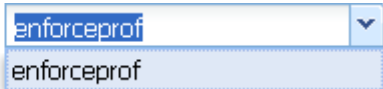
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.

5. Select a SIP interface service in the table and click **Edit**.
6. **Enforcement profile**—Choose the name of the enforcement profile you want to apply to the SIP interface from the combination box.

Enforcement profile

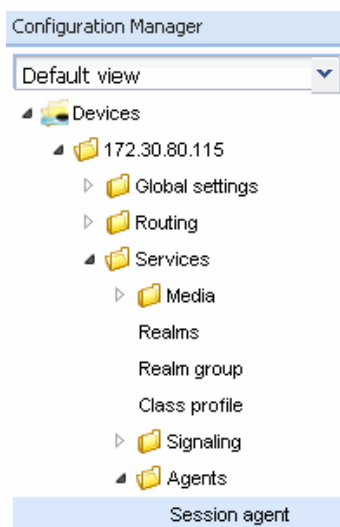


7. Click **Apply**.

Session Agent

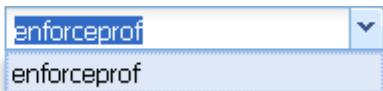
To apply an enforcement profile to a SIP session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. **Enforcement profile**—Choose the name of the enforcement profile you want to apply to the SIP session agent from the combination box.

Enforcement profile



4. Click **Apply**.

Local Policy Session Agent Matching

When you enable the local policy session agent matching option in your global SIP configuration, you change the way local policies match session agents. Normally, the Net-Net SBC looks up and stores matched session agents configured as next hops so it does not need to perform the lookup while processing requests. In this type of matching, the Net-Net SBC does take the realm set in the local policy attributes into consideration. When the Net-Net SBC performs its regular matching method and you have enabled overlapping IP addresses for session agents, the Net-Net SBC might match session agents to different realms than the ones you intended when creating your configuration.

Local policy session agent matching provides a way to match session agents differently, taking realms and nested realms into consideration during the matching process. This difference is key to deployments with multiple peering partners that use the overlapping IP address feature, and have multiple local policies routing to the same IP address in different realms where some target next hops require session constraints but others do not. In the cases where no session constraints are required, session agents are not needed. But session agents still match the local policy, applying their constraints, because they match the next hop IP address.

In addition to modifying this behavior, this feature also affects the use of realms and nested realms. It triggers the use not only of realms, but of all the realms nested however deeply—thereby improving matching efficiency.

How It Works

You can set the local policy session agent matching option with values that define how the Net-Net SBC performs session agent matching:

- **any**—The Net-Net SBC looks up and stores matched session agents configured as next hops so it does not need to perform the lookup while processing requests, without regard to realms.

This behavior is the default when the SIP configuration does not have the local policy session agent matching option set.

- **realm**—The Net-Net SBC selects session agents in the realm that the local policy attribute indicates; this provides an exact match, rather than not taking the realm into consideration during session agent selection.

For example, the session agent is a match if the session agent **realm-id** and the local policy attribute **realm** parameters are an exact match.

- **sub-realm**—Session agents in the same realm or the same realm lineage—where session agents and realms are related to one another via realm parent-child relationships no matter the depth of realm nesting configured

For example, the session agent is a match if the local policy attribute **realm** is a sub-realm of the realm specified in the session agent **realm-id** parameter.

- **interface**—Session agents in the same realm or same realm lineage via the realm set in the local policy attribute, and whose realm uses the same signaling interface as the realm set in the local policy attribute

For example, the session agent is a match if the session agent **realm-id** is a sub-realm of the local policy attribute **realm**, and both referenced realms use the same SIP signaling interface.

- **network**—Session agents whose realm is in the realm lineage for the same realm set in the local policy attributes, and whose realm is associated with the same network interface as the realm set in the local policy attributes

For example, the session agent is a match if the session agent **realm-id** is a sub-realm of the local policy attribute **realm**, and realm reference by both use the same network interface.

If it cannot find a match, the Net-Net SBC will use the IP address as the next hop. Further, requests matching local policy attributes will not be associated with session agents, and so their constraints will not be applied.

The Net-Net SBC stores session agent information that it looks up when performing local policing session agent matching. To perform the lookup, it uses the session agent hostname as a key. When the hostname is an FQDN and there is a configured

IP address, the Net-Net SBC uses the IP address as a secondary key. Given this implementation, the following are true when selecting session agents:

- If multiple session agents share the same IP address, the one with an IP address in the hostname parameter takes precedence.
- If all session agents with the same IP address have an FQDN as their hostname, the one whose name is alphabetically lower will take precedence, where “alphabetically lower” means earlier in the alphabet (closer to A than to Z).
- For non-global session agents (whose realms are configured but not wildcarded) with an IP address, the Net-Net SBC uses a key that is a combination of the IP address and the realm in the form <address>:<realm>.
- For a session agent whose realm has a parent realm, the Net-Net SBC uses a combination of the IP address, realm, and realm-path (or lineage for the realm) in the form <address>:<realm-path>. For example, the realm path for a realm **core3** with a parent **core2**, which in turn has a parent **core** would be **core:core2:core3**.

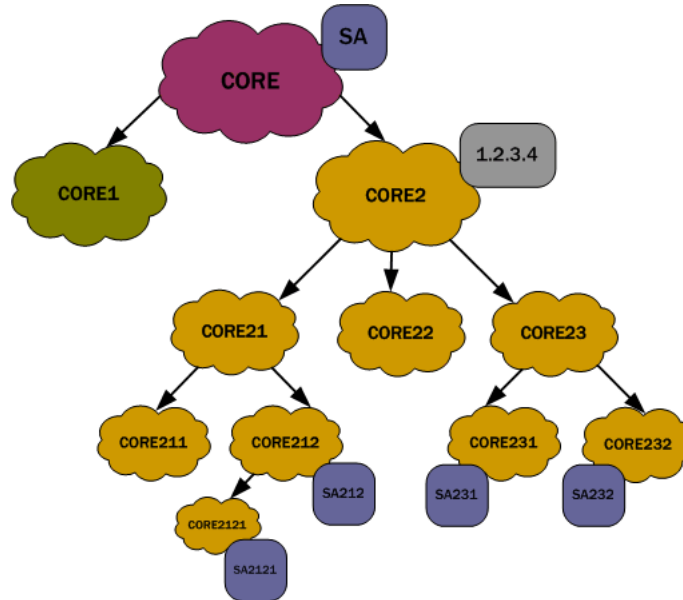
When it looks up a session agent with a realm, the Net-Net SBC first searches for an exact match for the IP address and realm combination. If this fails, it performs a second search if the desired realm has parents or children. The Net-Net SBC locates an entry in its repository of session agent information that is greater than or equal to the IP address with the base realm, which is the ancestor of the desired realm without a parent. Having gathered this set of candidates, the Net-Net SBC narrows down the search for a match by comparing sub-realms and determines there is a match if either:

- The desired realm path is a sub-string of the entry’s realm path, or
- The entry’s realm path is a substring of the desired realm path (i.e., the desired realm is a sub-realm of the entry’s realm)

Then the Net-Net SBC orders the candidates by depth of the entry’s realm-path, or number of levels from the base realm relative to the depth of the desired realm. By searching the ordered set until the entry’s realm depth equals the desired realm’s depth, the Net-Net SBC determines a parent candidate, all subsequent entries being sub-realms of the desired realm. The Net-Net SBC only considers entries at the first level deeper than the desired realm. If at this point there is only one entry, the Net-Net SBC deems it a match. Otherwise, it selects the parent candidate as the matching entry. In the event the search does not yield a matching realm, the Net-Net SBC uses the global session agent for the IP address, if there is one.

The following diagram shows the realm tree, where the clouds are realms and squares are session agents, representing a group of session agents sharing the IP

address 1.2.3.4. The Net-Net SBC searches for the session agents lower in the tree along the session agent realm-path and the desired realm.



For the diagram above, the following shows how the hostname would look for this group of session agents.

Key	Session Agent (hostname[realm])
1.2.3.4 (This session agent owns the primary key for the IP address because its hostname is the IP address.)	1.2.3.4[CORE2]
1.2.3.4:CORE (IP+realm key entry)	SA[CORE]
1.2.3.4:CORE (IP+realm key entry)	1.2.3.4[CORE2]
1.2.3.4:CORE212 (IP+realm key entry)	SA212[CORE212]
1.2.3.4:CORE2121 (IP+realm key entry)	SA2121[CORE2121]
1.2.3.4:CORE231 (IP+realm key entry)	SA231[CORE231]
1.2.3.4:CORE232 (IP+realm key entry)	SA232[CORE232]
1.2.3.4:CORE: (IP+realm-path key entry)	SA[CORE]
1.2.3.4:CORE:CORE2: (IP+realm-path key entry)	1.2.3.4[CORE2]
1.2.3.4:CORE2:CORE21:CORE212 (IP+realm-path key entry)	SA212[CORE212]

Key	Session Agent (hostname[realm])
1.2.3.4:CORE2:CORE21:CORE212:CORE2121 (IP+realm-path key entry)	SA2121[CORE2121]
1.2.3.4:CORE2:CORE23:CORE231 (IP+realm-path key entry)	SA231[CORE231]
1.2.3.4:CORE2:CORE23:CORE232 (IP+realm-path key entry)	SA232[CORE232]

For each realm in the table above, the search results for each realm would look like this:

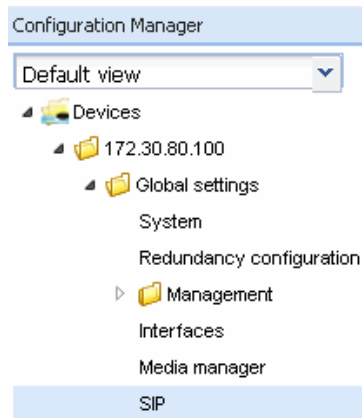
IP Address	Realm	Session Agent (hostname[realm])
1.2.3.4	CORE	SA[CORE]
1.2.3.4	CORE2	1.2.3.4[CORE2]
1.2.3.4	CORE21	SA212[CORE212]
1.2.3.4	CORE211	1.2.3.4[CORE2]
1.2.3.4	CORE212	SA212[CORE212]
1.2.3.4	CORE2121	SA2121[CORE2121]
1.2.3.4	CORE22	1.2.3.4[CORE2]
1.2.3.4	CORE23	1.2.3.4[CORE2]
1.2.3.4	CORE231	SA231[CORE231]
1.2.3.4	CORE232	SA232[CORE232]

Configuring Local Policy Session Agent Mapping

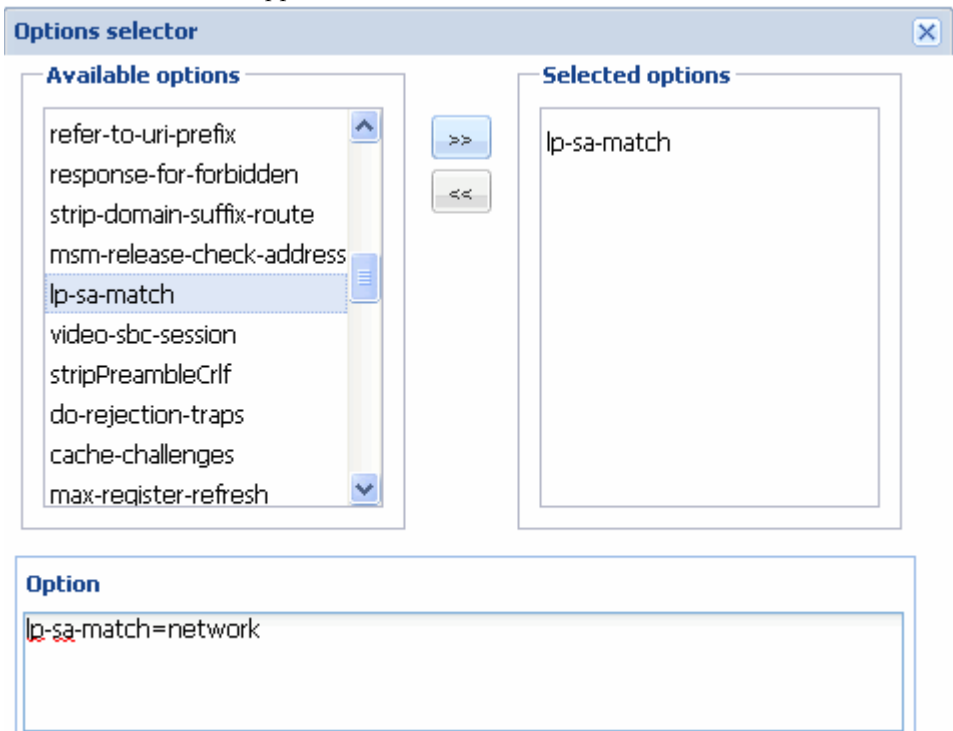
When you enable local policy session agent matching, remember that you can choose from five different ways to use the feature. See [How it Works](#) for details.

To enable local policy session agent matching using the realm method:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **lp-sa-match** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



7. **lp-sa-match**—Set the method for this option. The valid methods are: any, realm, sub-realm, interface, and network. For example:



If you do not specify a method, the Net-Net SBC uses the **all** method.

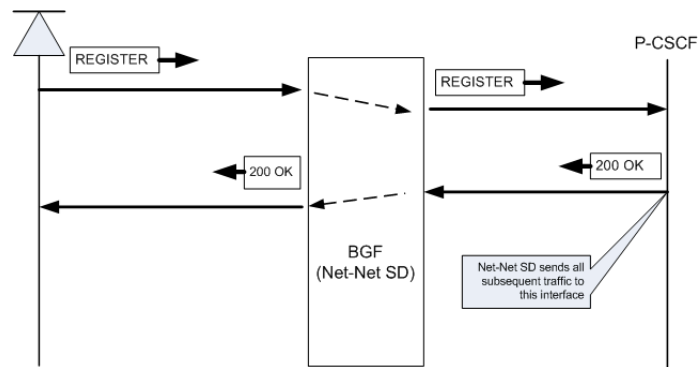
8. Click **OK**.
9. Click **Apply**.

Implicit Service Routing

In an IMS architecture, the Service-Route header indicates the network's requirement for where to direct all subsequent signaling from a given UE. Upon receipt of the Service-Route header(s) in a 200 OK response to a REGISTER request, the Net-Net SBC stores the values contained therein and preloads them when issuing new requests (such as an INVITE or a SUBSCRIBE) from the associated endpoint.

Although not all SIP networks fully support RFC 3608, some still require UEs to send all call traffic to the network element(s) with which they originally registered. Further, some P-CSCFs do not pass back the Service-Route headers in response to the UE's REGISTER request. This situation requires that a logical connection is made between the UE and the network element to which it registers.

In these scenarios, the Net-Net SBC may create an implicit Service-Route for the success (2xx class) response it receives from the P-CSCF. An implicit Service-Route is useful for creating a binding between a UE and a next-hop proxy, even when the Net-Net SBC can route to many potential, yet logically equivalent next-hops.

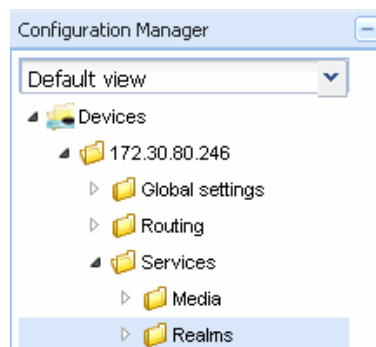


The Net-Net SBC's implicit Service-Route feature is configured on the access-side SIP interface where a UE registers.

Configuring Implicit Service Routing

To enable implicit Service-Route support:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.

4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll down to the Implicit service routing parameter.
7. **Implicit service routing**—Choose **enabled** from the combination box.

Implicit service routing

A screenshot of a web interface showing a dropdown menu for the 'Implicit service routing' parameter. The menu is open, displaying a list of options: 'disabled' (selected), 'replace', 'enabled', 'absent', 'strict', and 'disabled' (highlighted at the bottom). The dropdown is located to the right of the 'Implicit service routing' label.

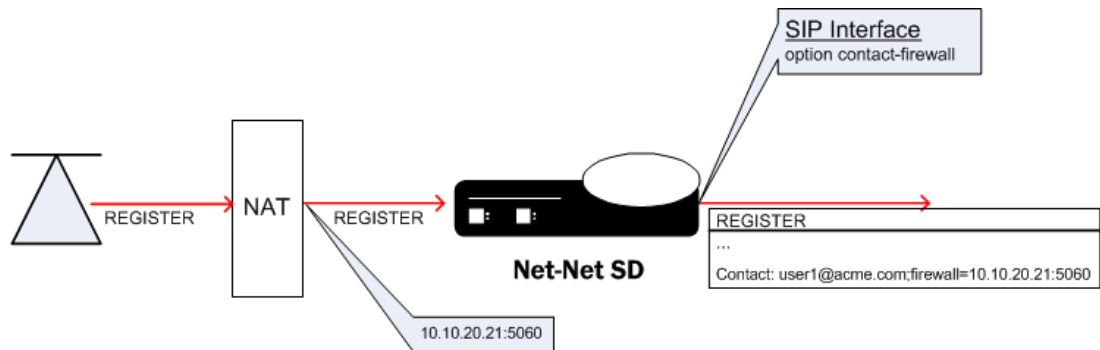
8. Click **Apply**.

Contact Header Contact Formatting

Some P-CSCF implementations require an extra parameter in the Contact header of the register message for full compatibility. This parameter should contain the public IP address of the UE.

The `contact-firewall` option is configured on the SIP interface attached to the core network.

When entering the `contact-firewall` option without a value, the SD will add a parameter to the Contact header called `firewall`. The value will be `IPaddress:port`. Note the appearance of the contact header in the following example and how it includes the endpoint's public IP address.

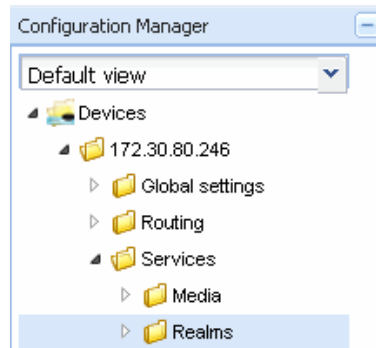


To change the parameter name from "firewall" to a name of your choosing, enter the option as `contact-firewall="parameter_name"`.

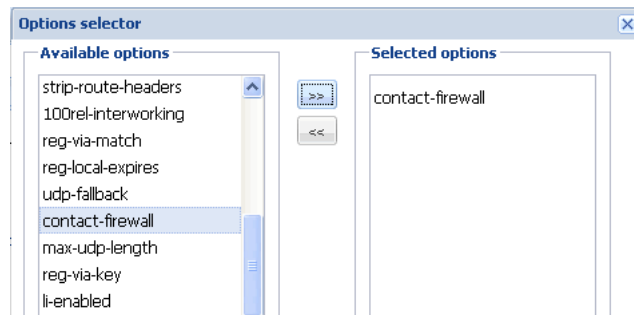
Configuring Contact Header Contact Formatting

To add a UE's public IP address and port to the contact header:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **contact-firewall** in the Select options list to select it.



8. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
9. Click **OK**.
10. Click **Apply**.

403 Response Code Override

You can override the actual response code sent back to an endpoint when conditions dictate that a 403 Forbidden response be sent. This is accomplished by adding the `response-for-forbidden=<value>` option, where value = response code number you want sent back to the endpoint. The 403 override is configured in the global SIP configuration.

Configuring 403 Response Code Override

To override the 403 Forbidden response code:

1. In the Configuration slider, change the Default view to ACLI view.
2. Expand the session-router folder.
3. Click sip-config.
4. Scroll to options and click **Options**. The Options selector dialog box appears.
5. Click **response-for-forbidden** in the Available options list to select it.
6. Click  to move the option to the Selected options list. A description of this option appears in the Description field.
7. Type =403 in the in the Option field.
8. Click **OK**.
9. Click **Apply**.

SIPconnect

The Net-Net SBC supports the SIPconnect model, wherein PBXes register themselves so that service providers do not need to know IP addresses or locations in advance for static configurations. This is particularly helpful when the PBX is behind a NAT.

In the PBX registration process, the PBX creates a binding between one of its phone numbers as the address of record (AoR) and Contact-URI in the REGISTER message. The registrar knows that the single AoR actually represents many addresses, and so it registers them implicitly. However, the registrar does not return the implicit AoR number in P-Associated-URIs.

The SIPconnect feature resolves the following issues that arise from using this model:

- SIP INVITEs sent to the PBX from the Registrar through the Net-Net SBC have the Request-URI of registered contact. Because it typically ignores the To-URI, the PBX needs the Request-URI username portion to be the specific extension number being called.

With the SIP connect feature enabled, the Net-Net SBC overwrites the Request-URI username with the To-URI username.

- SIP INVITEs from the PBX have the From AoR and Contact-URI usernames of specific phones rather than of the registered AoR and Contact-URI. For the Net-Net SBC, this means that it cannot use the **Anonymous mode** parameter value of **register**; there would be no registered user matches, and the Net-Net SBC would reject them (with a 403 Forbidden).

With the SIP connect feature enabled, the Net-Net SBC performs allow-anonymous checking based on the registered Via address, which is the same for all requests for the same PBX.

Modifications to Registration Caching Behavior

With the SIPconnect feature enabled, Net-Net SBC registration caching works the same way that it does with the feature disabled, with the following exceptions:

The Net-Net SBC determines whether the destination realm has the `sip-connect-pbx-reg` option configured, and then:

- If it is configured, the Net-Net SBC replaces the user part of the Request-URI with the user part of the To header. When the INVITE contains a P-Called-Party-ID header, the Net-Net SBC uses the user part of the P-Called-Party-ID header (instead of the To header).
- If it is not configured, the Net-Net SBC determines if the destination address is for a session agent and whether that session agent has `sip-connect-pbx-reg` option configured. When it is configured, the Net-Net performs the same replacements described in the bullet directly above. When it is not configured, the Net-Net SBC does not make any replacements.

When it receives an INVITE request, the Net-Net SBC checks the incoming realm for the `sip-connect-pbx-reg` option.

- If it is configured, the Net-Net SBC uses the INVITE's source address (instead of the AoR and Contact-URI) to search the registration cache for a matched registration entry.
- If it is not configured, the Net-Net SBC determines if the INVITE's source address is for a session agent and whether that session agent has `sip-connect-pbx-reg` option configured.

When it is configured, the Net-Net SBC replaces the user part of the Request-URI with the user part of the To header. When the INVITE contains a P-Called-Party-ID header, the Net-Net SBC uses the user part of the P-Called-Party-ID header (instead of the To header).

When it is not configured, the Net-Net SBC does not make any replacements.

Configuring SIPConnect Support

You configure this feature by adding the `sip-connect-pbx-reg` option to the realm configuration. In addition, this feature requires that your configuration also be set up as outlined in this section. The first two items are required, and Acme Packet recommends that you also implement the suggested additional configuration.

Required Configuration

- Registration caching is enabled.
- For the realm from which registrations come, the options list must include `sip-connect-pbx-reg`; this is new configuration introduced to support this feature. The presence of this option instructs the Net-Net SBC to skip matching the Contact header in the INVITE request with the registered Contact of the registration entry. The Net-Net SBC finds a registration using only the INVITE's source address.

Alternatively, you can configure the `sip-connect-pbx-reg` option in the options list for a session agent. When the realm where an INVITE comes from does not have this option set, the Net-Net SBC determines whether or not the INVITE came from a session agent. You might choose to configure session agents with this option if you do not want it applied to an entire realm. If the PBX is behind a NAT device, the session agent's IP address for the PBX (if statically configured)

must be the IP address of the NAT device. And if DNS is use, the session agent's hostname must resolve to the NAT device's IP address.

Suggested Additional Configuration

- In the SIP ports configuration (accessed through the SIP interface configuration), the **Anonymous mode** parameter must be set to **registered**. This setting allows the Net-Net SBC to accept SIP requests from session agents and registered endpoints only, but to accept REGISTER requests from any endpoint.
- For the SIP interface that accepts registrations, set the **reg-via-key** option. This setting allows the Net-Net SBC to use the source address of an INVITE as the key to find a registration entry in the registration cache. When the INVITE's Contact header matches the registered Contact in the registration entry, the Net-Net SBC accepts the INVITE request.

Configuring SIPConnect

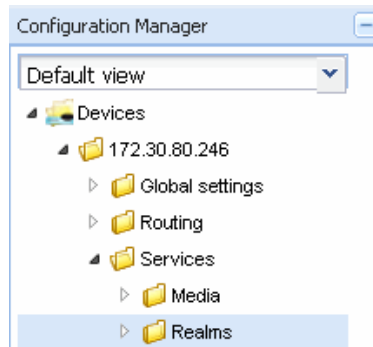
You can configure SIPconnect for:

- realms
- session agents

Realm

To set the SIPconnect option:

1. Expand the Configuration Manager slider > Services.

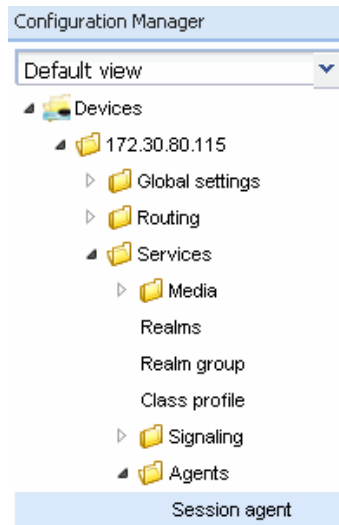


2. Click Realms. The Realm table appears in the content area.
3. Click the realm to which you want to configure in the table and click **Edit**.
4. Click **Options**. The Options dialog box appears.
5. Click **sip-connect-pbx-reg** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
7. Click **OK**.
8. Click **Apply**.

Session Agent

To set the SIP connect option:

1. Expand the Configuration Manager slider > Services > Agents.

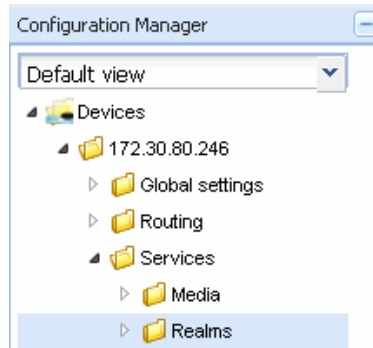


2. Click Session agent. The Session agent table appears in the content area.
3. Click **Options**. The Options dialog box appears.
4. Click **sip-connect-pbx-reg** in the Available options list to select it.
5. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
6. Click OK.
7. Click Apply.

Configuring the SIP Interface

To configure the SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **reg-via-key** in the Available options list to select it.
8. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
9. Click **OK**.
10. Click **Apply**.

SIP Port Mapping Enhancement

When you enable SIP port mapping in pre-C6.0.0 releases, the Net-Net SBC allocates a signaling port from a configured pool as it creates the registration cache entry for a Contact in a REGISTER request. For each unique Contact URI, the Net-Net SBC allocates a separate signaling port. This registration cache Contact entry contains the:

- Mapping between the Contact URI in the access-endpoint realm (or the UA-Contact)
- Contact URI in the registrar-softswitch realm (or the SBC-Contact), which is then used as the signaling port

Each PBX registration has its own port number on the Net-Net SBC.

However, using SIP port mapping with SIPconnect—where unique ports are used for each registered PBX—prevents the Net-Net SBC from routing incoming calls to the corresponding PBX. These instances occur because the Net-Net SBC uses directory number (DN) for the PBX's parent during registration, but the incoming INVITE from the softswitch contains the child DN in its Request URI. Thus the Net-Net SBC cannot find a matching SBC-Contact because the username of the Request URI contains the child DN, but the username of the SBC-Contact contains the parent DN.

An enhancement to SIP port mapping lets the Net-Net SBC matches SIPconnect registration entries exclusively on host-port information. It ignores the username.

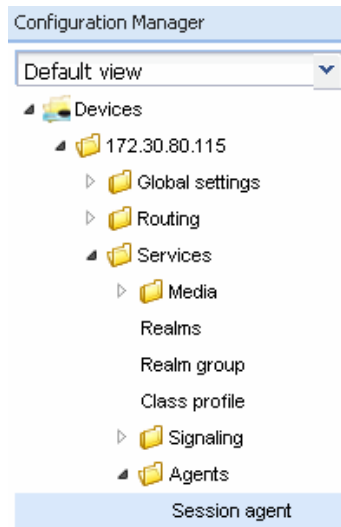
How It Works

You can enable SIPconnect support in either the realm configuration or session agent for the SIP access network by setting the **sip-connect-pbx-reg** option. With this option set and the destination realm configured for port mapping, the Net-Net SBC inserts a special search key in the registration table. Rather than adding the SD-Contact as the key as with regular (non-SIPconnect) registrations, the Net-Net SBC strips user information and instead uses the host and port information as the registration key. The Net-Net SBC still—as it would in pre-C6.0.0 releases—forwards the registration message with an intact contact username.

Configuring SIPconnect Support

To configure SIPconnect support from session agents:

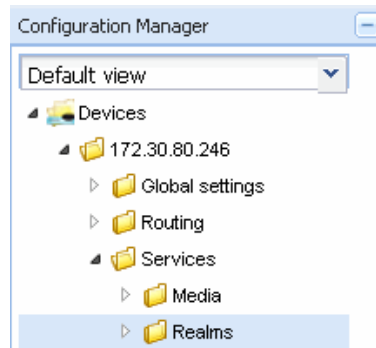
1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click **Options**.
4. Click **sip-connect-pbx-reg** from the list of Available options to select it.
5. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
6. Click **OK**.
7. Click **Apply**.

To configure SIPconnect support from realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click a SIP interface realm in the table and click **Edit**.
4. Click **Options**.
5. Click **sip-connect-pbx-reg** from the list of Available options to select it.
6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
7. Click **OK**.
8. Click **Apply**.

HNT Contact Endpoint

You can now exclude the port number from an endpoint URL when that endpoint is behind a NAT. This is an enhancement to the SIP option **contact-endpoint**. Using that option caused the Net-Net SBC to add a URL parameter (defined as an argument to the contact-endpoint) to the Contact headers of REGISTER messages that it forwarded to the registrar when it performed registration caching.

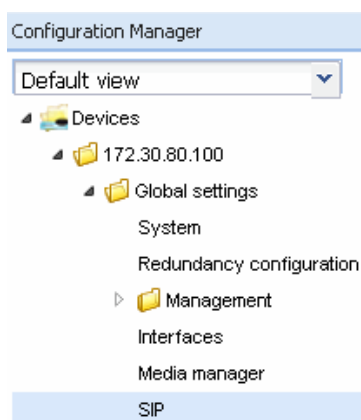
The value of the contact-endpoint URL parameter was the real address of the endpoint; and if the endpoint was behind a NAT, this included the IP address and a port number. However, not all network entities can parse that port number. This new feature provides a mechanism for excluding it.

Configuring HNT Contact Endpoint

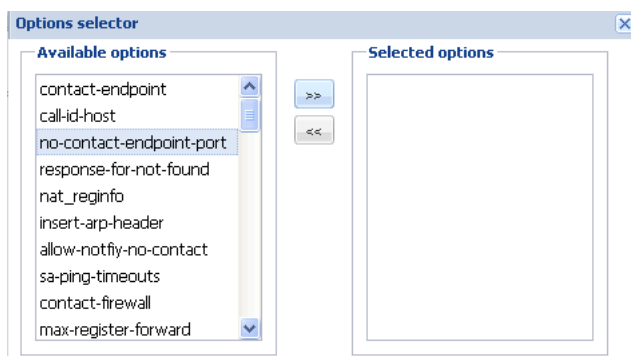
You enable this feature for the global SIP configuration by setting **no-contact-endpoint-port=yes**. Although you set this parameter in the global SIP configuration, it is applied only to SIP interfaces.

To enable SIP HNT contact endpoint:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.
5. Click **no-contact-endpoint-port=yes** in the Available options list to select it.



6. Click **>>** to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.

7. Click **OK**.
8. Click **Apply**.

Delayed Media Update

The Net-Net SBC supports SIP delayed media update. When enabled, this feature keeps the Net-Net SBC from updating its media flow information for flows established after an offer-answer exchange. The Net-Net SBC does not update the flow information until a new offer and answer arrive for a specific set of media flows. The (subsequent) offer does not have to be for the same session; rather, it can appear as a new SIP INVITE that uses the same SDP.

Previous Behavior

In prior releases, the Net-Net SBC updates media flow entries in its CAM based on signaled SDP when it processes the SDP. If it processes an SDP offer, the Net-Net SBC allocates steering port resources and updates any missing elements for the flow when the answer is returned.

In cases when a secondary offer arrives (either a reINVITE, an UPDATE, or the original INVITE is hairpinned back through the Net-Net SBC), the Net-Net SBC updates the following media flow information at the time of the offer:

- Destination IP address
- Destination port
- Realm for the media flows
- Media release settings

This behavior affects specific applications that are better served by the Net-Net SBC waiting to update media flow information until it receives the answer to the second offer.

New Behavior

When you enable SIP delayed media, the Net-Net SBC:

- Delays changing the active media flow CAM entry for a new offer if a previous offer and answer have been received for the same media flows; it encodes new SDP information in an outgoing offer, but does not change the CAM entry until the answer is received
- Delays changing the active media flow CAM entry even when the new offer is for a new session
- Supports media release when performing delayed media update changes
- Offers per-realm configuration

How It Works

This section describes how the delayed media update feature works for hairpinned call flows and for an SDP offer arriving for installed flows.

- **Hairpinned call flows**—In this type of call flow, the application server (AS) sends an INVITE back to the Net-Net SBC and that INVITE needs to be forwarded to another user (user B). When it receives the offer in this INVITE and delayed media update is disabled, the Net-Net SBC determines that the call is hairpinned and deletes the CAM entry for the flow for user A, who has sent the initial INVITE. The Net-Net SBC deletes the CAM entry for the flow from the AS to user A.

With delayed media update enabled, the CAM entry for the flow from the AS to user A is not deleted. Instead, the Net-Net SBC waits until it has an answer from user B, and then performs the necessary updates and deletions.

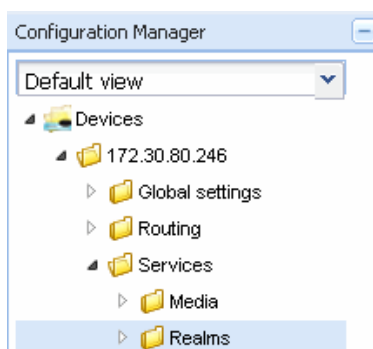
- **SDP offer for installed media flows**—With delayed media update enabled, if it has received an offer and answer and a new offer arrives for the same flow, the Net-Net SBC delays updating the CAM entries until an answer is received for the new offer.

Configuring Delayed Media Update

You enable this feature on a per-realm basis by setting one parameter.

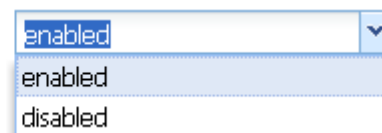
To enable SIP delayed media update:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click a SIP realm in the realm table to select it and click **Edit**.
4. **Delayed media update**—Select enabled from the combination box to enable it.

Delay media update



5. Click **Apply**.

Interface Response Code Mapping

You can configure SIP interfaces on your Net-Net SBC to use SIP response maps, which were previously only applicable to session agents.

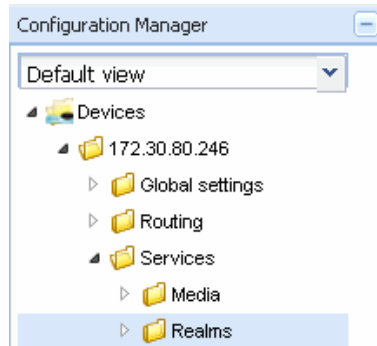
For each SIP interface, the Net-Net SBC supports two kinds of response code maps: one for SIP responses received from other devices, and a second for mapping internally-generated responses. The Net-Net SBC applies these responses to the entity the response are sent to. So when you apply them to a SIP interface, they map the responses sent out that SIP interface; when you apply them to a session agent, they map the responses sent to that session agent.

Configuring SIP Interfaces

This section describes the two parameters that you can use in the SIP interface configuration to use one response code map for incoming SIP responses and one for internally-generated responses.

In each case, the value you enter is the name of a SIP response map configuration. Be sure to type the name correctly; otherwise the Net-Net SBC will not perform response code mapping.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click the Registration at the top of the content area and scroll to the SIP response map sections.
7. **Response code translation profile**—Choose the name of the SIP response map configuration that you want to apply to this SIP interfaces for outgoing responses from the combination box.

SIP response map

Response code translation profile	sipresponse1
Local response code translation profile	sipresponse1

8. **Local response code translation profile**—Enter the name of the SIP response map configuration that you want to apply to this SIP interfaces for locally-generated SIP responses from the combination box.
9. Click **Apply**.

Port Mapping Based on IP Address

SIP port mapping is a feature that enables the Net-Net SBC to represent registered SIP endpoints on the core side using unique SIP UDP ports. It provides the ability for the Net-Net SBC to map ports based on the address of record (AoR) and Contact address of the endpoint. This way, the registrar sees each endpoint's registration as coming from a unique port number. Then the registrar expects to receive subsequent requests from those ports on the Net-Net SBC; it also sends messages destined for a specific endpoint to the correct port on the Net-Net SBC. If the same endpoint registers with two contacts (for two AoRs), the Net-Net SBC allocates two unique core-side ports for that endpoint.

However, some registrars need to know that multiple contacts represent the same endpoint. The extension to this feature answers the expectation from registrars that an endpoint registering multiple AoRs will use a single core-side mapped port to show that the AoRs really represent a single endpoint.

When you enable this new SIP port mapping capability, the Net-Net SBC supports core-side UDP port mapping based on the endpoint's IP address. It ignores the username portion of the AoR or Contact.

How It Works

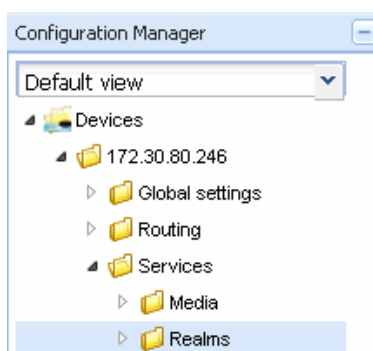
When you enable SIP port mapping based in IP address, the Net-Net SBC performs the port mapping allocation and lookup based on all requests using the via-key from the SIP Request. The via-key is a combination of Layer 3 and Layer 5 IP information in the message. The Net-Net SBC performs an additional lookup in the registration table to determine if a via-key already exists. If it does, then the Net-Net SBC uses the port already allocated and does not allocate a new one.

Configuring Port Mapping

Modifications to the pre-existing **reg-via-key** option have been made to support this feature. To use SIP port mapping without the IP address enhancement, enter **reg-via-key**. If you want to enable SIP port mapping based on IP address, you need to configure the option as **reg-via-key=all**.

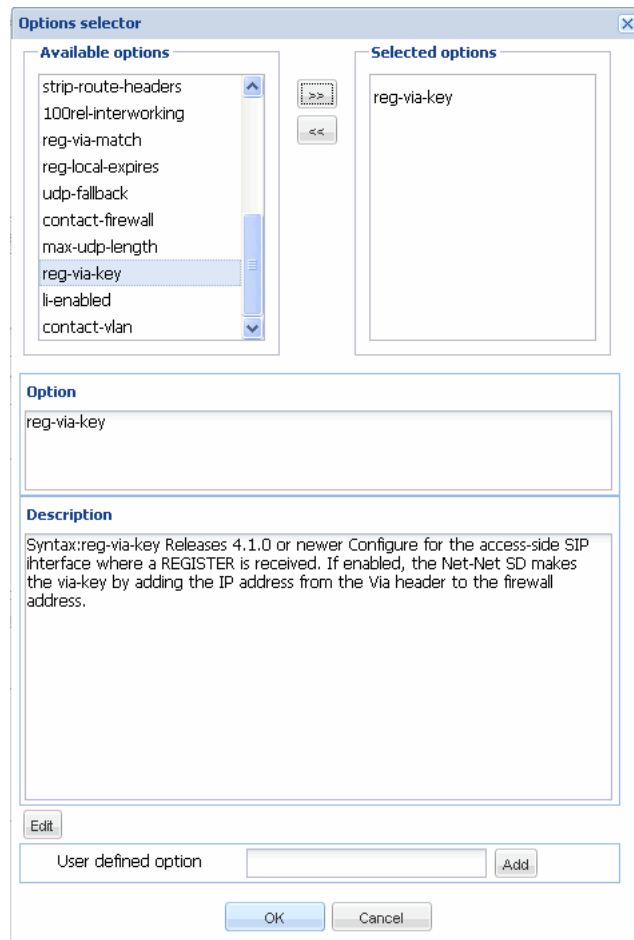
To enable SIP port mapping based on IP port:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.

6. Scroll to Options in SIP service and click **Options**. The Options dialog box appears.
7. Click **reg-via-key** in the Available options list to select it.
8. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.



9. Click **OK**.
10. Click **Apply**.

Registration Overload Protection

You can configure your Net-Net SBC for SIP registration overload protection, which augments the Net-Net SBC's pre-existing protection methods. Working with the Net-Net SBC's access control and registration caching functions, this feature guards against benign avalanche restarts. The avalanche is caused by events where many endpoints lose power or connectivity at once, are restored to service, and then flood the Net-Net SBC as they attempt to register again.

How It Works

Normally, the Net-Net SBC handles SIP registration by creating a temporary registration cache for the endpoint's address of record (AoR) and forward the REGISTER request to the registrar. To challenge the endpoint's registration, the registrar sends back either a **401 Unauthorized** or **407 Proxy Authorization Required** response. When it receives the 401 or 407, the Net-Net SBC saves the challenge context in anticipation of receiving a second REGISTER with the endpoint's authentication credentials. The Net-Net SBC forwards the second REGISTER (with authentication credentials) to the registrar, and then the registrar confirms registration with a 200 OK. Both REGISTER requests are subject to the Net-Net SBC's access control rules, set either for the ingress realm or the ingress session agent. The Net-Net SBC also honors the maximum registration sustain rate constraint for session agents; this applies when the incoming REGISTER is from a session agent and the outgoing REGISTER is sent to a session agent.

When you enable SIP registration overload protection, the Net-Net SBC temporarily promotes the endpoint to the trusted level when it receives the 401 or 407 response (to the first REGISTER) from the registrar. This ensures that the second REGISTER (containing authentication credentials) can reach the Net-Net SBC. Temporary promotion lasts only for the amount of time remaining before the REGISTER server transaction expires plus the time allotted in the transaction expiration parameter in the SIP configuration. Before the temporary promotion expires, there is enough time for any necessary retransmission of the first REGISTER and for the second REGISTER to take place. The following situations might also occur:

- If the Net-Net SBC receives a 401 or 407 to the second REGISTER request, it resets its access control level for the endpoint's address to the default level; it then treats additional REGISTER requests from the same context at the default access control level.
- If the Net-Net SBC receives a 200 OK response to the REGISTER message, it extends the promotion time to the expiration period for the registration cache.

If the Net-Net SBC is unable to find the temporary registration cache and the saved challenge context when the second REGISTER arrives, it forwards the REGISTER without checking the maximum registration sustain rate constraint for ingress and egress session agents—thereby ensuring that the REGISTER with authentication credentials is sent to the registrar.

When you use this feature, you should set the maximum registration sustain rate constraint of the session agent (representing the registrar) at half the registrar's maximum registration sustain rate. Additional REGISTER requests with the same challenge context are subject to the maximum registration sustain rate constraint.

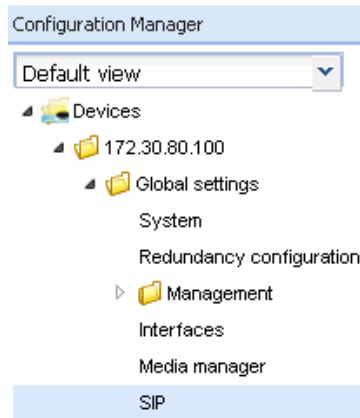
Configuring SIP Registration Overload Protection

You set the following two options in your global SIP configuration to configure this feature:

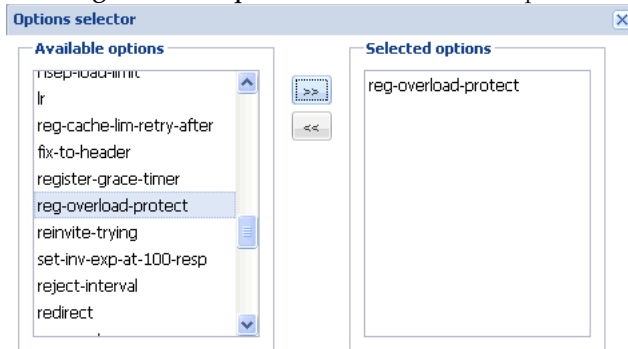
- **reg-overload-protect**: causes the Net-Net SBC to temporarily promote an endpoint to the trusted level when it receives the 401 or 407 response
- **cache-challenges**: required so that SIP registration overload protection can recognize an INVITE request that has authentication and exempt it from SIP registration overload protection

To enable SIP registration overload protection:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.
5. Click **reg-overload-protect** in the Available options list to select it.



6. Click **>>** to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
7. Repeat steps 5 and 6 to select and move the cache-challenges option.
8. Click **OK**.
9. Click **Apply**.

SIP Request Method Throttling

You can configure throttling mechanisms for some types of SIP methods using the rate constraints configuration available both in the session constraints (which you then apply to a SIP interface or a realm) and the session agent configurations.

Acme Packet recommends you use session agent constraints for session rate INVITE throttling and registration rate for REGISTER throttling.

For SIP access deployments, you can configure rate constraints for individual method types along with a set of burst and sustain rates. These constraints can help to avoid overloading the core network. In addition, they restrain the load non-INVITE messages use, thus reserving capacity for INVITE-based sessions and Registrations

How It Works

When you configure SIP request method throttling, ensure you do not reject in-dialog requests. Acme Packet recommends you do NOT configure constraints for the following SIP method types:

- ACK
- PRACK
- BYE
- INFO
- REFER

The Net-Net SBC is likely to throttle NOTIFY requests despite their being part of a Subscribe dialog.

The methods you will most likely configure for throttling are:

- NOTIFY
- OPTIONS
- MESSAGE
- PUBLISH
- REGISTER

The Net-Net SBC counts Re-INVITEs and challenges responses against the throttle limit, but does not check to determine if the constraints have been exceeded for either.

You can configure separate constraints—inbound and outbound values for burst and sustain rates—for each different method type you configure. Although you should use session agent constraints (and not rate constraints) for INVITEs, if you also set up rate constraints for INVITEs, then the smallest configured value takes precedence.

About Counters and Statistics

Each rate constraint you configure for a SIP method tracks its own counters. You can set the burst rate window in the session constraints configuration that will apply to all methods configured as rate constraints.

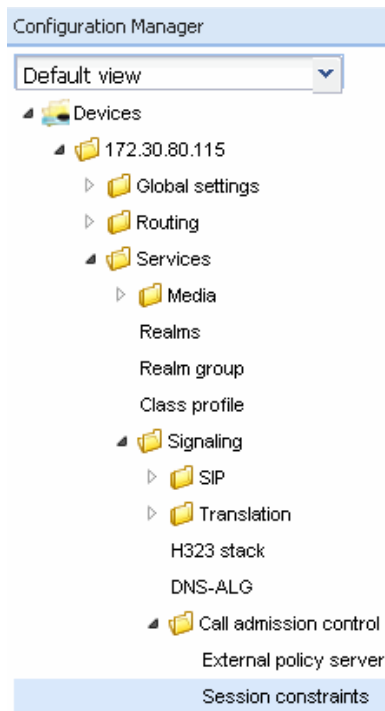
The Net-Net SBC captures statistics for SIP methods throttled by rate constraints for SIP interfaces and session agents; it does not capture these statistics for the global SIP configuration.

Configuring Rate Constraints for Session Constraints

This section shows you how to set up rate constraints for session constraints, which are then applied to SIP interfaces.

To configure rate constraints for session constraints:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click Session constraints.
3. Select a session constraint in the table and click **Edit**.
4. Scroll to Rate constraint and click **Add**. The Add Rate constraint window appears.

5. **SIP method**—Click a SIP method name from the combination box for the method you want to throttle.

6. Click **Apply**.
7. **Max inbound burst rate (bps)**—Enter the number in bytes per second to restrict the inbound burst rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.
8. **Max inbound sustained rate (bps)**—Enter the number in bytes per second to restrict the inbound sustain rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.
9. **Max outbound burst rate (bps)**—Enter the number in bytes per second to restrict the outbound burst rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.
10. **Max outbound sustained rate (bps)**—Enter the number in bytes per second to restrict the outbound sustain rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.

*SIP method	OPTIONS ▼
Maximum inbound burst rate(bps)	0
Maximum inbound sustained rate(bps)	0
Maximum outbound burst rate(bps)	0
Maximum outbound sustained rate(bps)	0

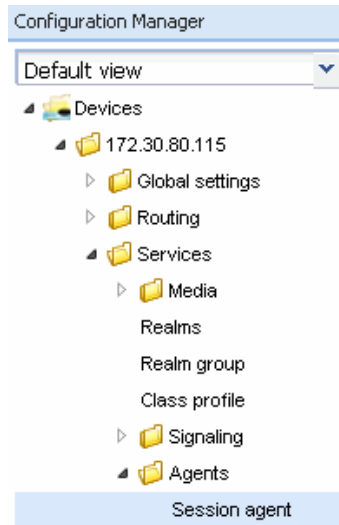
11. Click **Apply**.
12. Click **Apply**.

Configuring Rate Constraints for SIP Session Agents

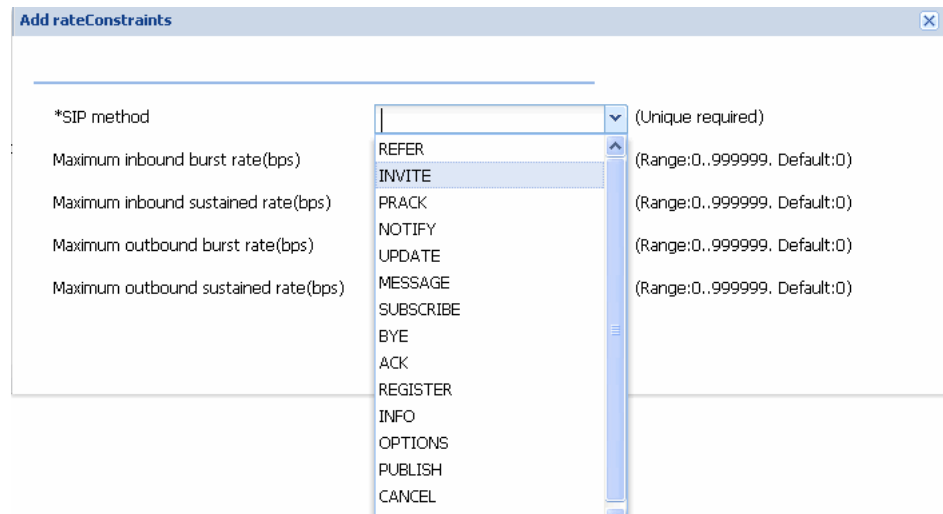
This section shows you how to set up rate constraints for individual SIP session agents.

To configure rate constraints for SIP session agents:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the Constraints tab at the top of the content area.
4. Scroll to the Rate constraint table and click **Add**. The Add SA Rate constraint window appears.
5. **SIP method**—Click a SIP method name from the combination box for the method you want to throttle.



6. **Max inbound burst rate (bps)**—Enter the number in bytes per second to restrict the inbound burst rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.
7. **Max inbound sustained rate (bps)**—Enter the number in bytes per second to restrict the inbound sustain rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.
8. **Max outbound burst rate (bps)**—Enter the number in bytes per second to restrict the outbound burst rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.
9. **Max outbound sustained rate (bps)**—Enter the number in bytes per second to restrict the outbound sustain rate on the SIP interface where you apply these constraints. The default and minimum value is 0, and the maximum is 999999999.
10. Click **Apply**.
11. Click **Apply**.

SIP Registration Response Mapping

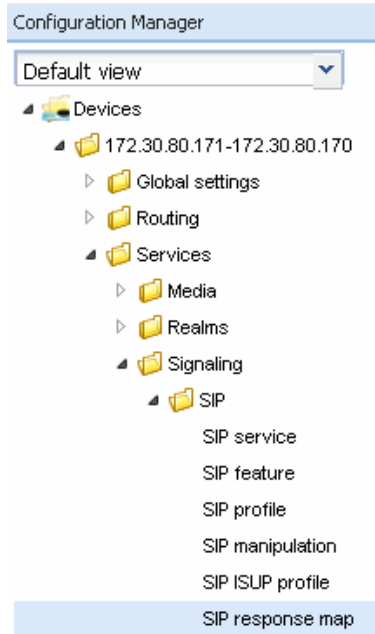
SIP registration response mapping allows you to configure the Net-Net SBC to remap a SIP failure response—which it receives from another network device or that it generates locally—to a 200 OK. You might want the Net-Net SBC to perform this type of mapping for circumstances where non-malicious endpoints continually attempt registration, but will stop (and still not be registered) when they receive a 200 OK. This response mapping does not actually register the client with the Net-Net SBC, meaning that there is neither a registration cache entry or a CAM ACL for it.

For the 200 OK it generates, the Net-Net SBC removes any Reason or Retry-After header in the 200 OK and sets the expires time. By default, the expires time is the Retry-After time (if there is one in the response) or the expires value in the Register request (if there is no Retry-After expires time). You can also set this value using the register-response-expires parameter, but the value you set should never exceed the Register request's expires time.

Configuring SIP Registration Response Mapping: SIP Interface

To configure SIP registration response mapping for a SIP interface:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP response map. The SIP response map table appears in the content area.
3. Click a SIP response profile name in the table and click **Edit**.
4. Click Add in the Response code table.
5. Select a Response code translation from the table and click **Edit**.
6. **Received SIP response code**—Enter the original SIP response code received.
7. **SIP Response code**—Enter the SIP response code.

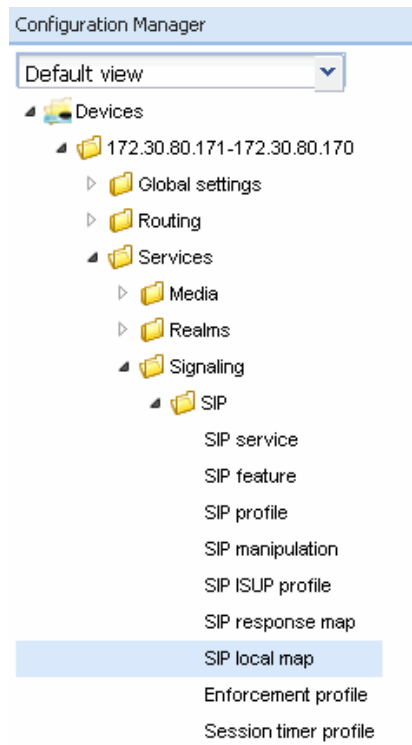
The screenshot shows the 'Add Response code translation' dialog box. It contains two input fields, both with the value '100'. The first field is labeled '*Received SIP response code' and has a tooltip that says '(Unique required. Range:100..699. Default:100)'. The second field is labeled '*SIP response code' and has a tooltip that says '(Range:100..699. Default:100)'. At the bottom of the dialog are 'Apply' and 'Cancel' buttons.

8. Click **Apply**.

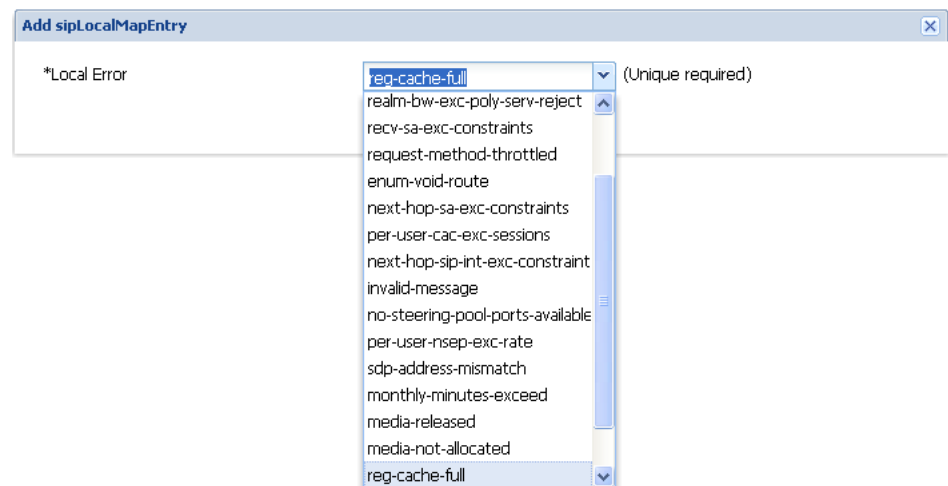
Configuring SIP Registration Response Mapping: Local Response Map

To configure SIP registration response mapping for a local response map:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



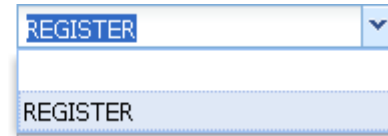
2. Click SIP local map.
3. Click **Add** to add the configuration. The SIP local map table displays in the content area.
4. **Local error**—Select the local error that triggers the use of this local response map from the combination box.



5. Click **Apply**.
6. Select the local map from the table and click **Edit**.

7. **Method**—Click the name of the received SIP failure response message you want to map to a 200 OK from the combination box. There is no default for this parameter, and leaving the parameter empty turns off the SIP registration response mapping feature.

Method



8. **Register response expires(sec)**—Enter the time in seconds you want to use for the expires time when mapping the SIP method you identified in the SIP method from step 5. The maximum is 999999999. By default, the expires time is the Retry-After time (if there is one in the response) or the expires value in the Register request (if there is no Retry-After expires time). Any value you configure in this parameter (when not using the defaults) should never exceed the Register request's expires time.

Register response expires(sec)

30

9. Click **Apply**.

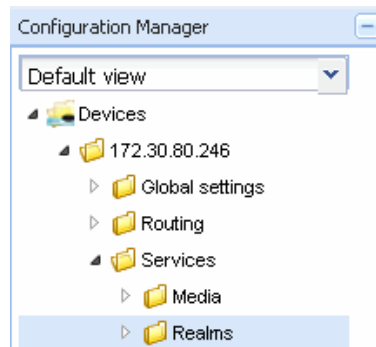
Configurable Timers and Counters

In prior releases of the Net-Net OS, SIP timers and counters are only configurable globally. This new release gives you the ability to configure timers for each SIP interface for greater flexibility. You can set the expiration times for SIP messages, and you can set a counter that restricts the number of contacts that the Net-Net SBC tries when it receives a REDIRECT.

Configuring Timers and Counters

To configure timers and counters for SIP interfaces:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to the Configurable timers and counters section.

7. **Transaction expire/timer F (secs)**—Enter the TTL in seconds for SIP transactions. This timer controls the following timers specified in RFC 3261:
 - Timer B—SIP INVITE transaction timeout
 - Timer F—non-INVITE transaction timeout
 - Timer H—Wait time for ACK receipt
 - Timer TEE—Used to transmit final responses before receiving an ACK

You can leave this parameter set to the default value of 0 or enter a value up to 999999999. If you set this parameter to 0, the Net-Net SBC uses the timer value from the global SIP configuration.

8. **INVITE expire/timer B (secs)**—Enter the TTL in seconds for a SIP client/server transaction after receiving a provisional response.

You set this timer for the client and the sever by configuring it on the SIP interface corresponding to the core or access side.

You can leave this parameter set to the default value of 0 or enter a value up to 999999999. If you set this parameter to 0, then the Net-Net SBC uses the timer value from the global SIP configuration.

9. **Max REDIRECT contacts**—Enter the maximum number of contacts or routes for the Net-Net SBC to attempt when it receives a SIP Redirect (3xx Response). If you leave this parameter set to the default value of 0, the Net-Net SBC will not exercise restrictions on the number of contacts or routes. The value range for this parameter is 0 to 10.

Configurable timers and counters

Transaction expire/timer F (sec)	0
INVITE expire/timer B (sec)	0
Max. REDIRECT contacts	0

10. Click **Apply**.

Mid-Call Re-Registration Following Service Route

A new `idle` value has been developed for the `route-register-no-service-route` SIP option to ensure that, when enabled, the Net-Net SBC follows the previously defined rules for idle calls, where `idle` means not engaged in any Invite-based sessions.

In prior releases, the `route-register-no-service-route [all | refresh]` option controlled how a User Agent (UA) is re-registered. The following are definitions of the option and its arguments:

- **route-register-no-service-route**—Prevents the use of the Service-Route procedure to route the Re-Register requests after the UA has initially registered.
- **route-register-no-service-route=all**—Prevents the use of the Service-Route procedure to route the Re-Register requests for all messages, after the UA has initially registered.

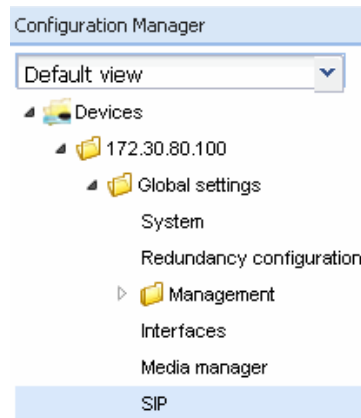
- **route-register-no-service-route=refresh**—Prevents the use of the Service-Route procedure to route the Re-Register requests for all refresh-register messages, but not de-register messages, after the UA has initially registered.

For some users, however, depending on their network, re-registering UAs while a call is taking place can cause problems and allow the call to be torn down.

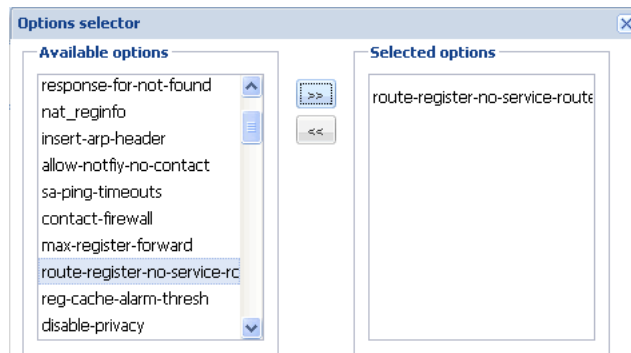
Configuring SIP Mid-Call Re-Registration

To configure SIP mid-call re-registration:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.
5. Click **route-register-no-service-route** in the Available options list to select it.



6. Click  to move the option to the opposite list. The option name appears along with a description in the lower part of the window.
7. **route-register-no-service-route**—Edit the textbox to append a label for the information. For example:

`route-register-no-service-route=idle`

8. Click **OK** to save your edits and close the Options dialog box.
9. Click **Apply**.

Random Registered-Contact

The SIP random registered contact feature lets the Net-Net SBC support endpoints that randomly change their contact username every time they re-register. This feature is only applicable to operators who need to support the Japan TTC standard JJ-90.22 in specific applications.

This feature applies to cases when an endpoint re-registers with a different contact username, but with the same hostname/IP address and the same address of record (AoR). Without this feature enabled, the Net-Net SBC forwards every re-registration to the registrar with the new contact information without it being considered a registration refresh. The Net-Net SBC forwards it to the Registrar using the same sd-contact as the previous registration.

When you enable this feature, however, the Net-Net SBC does treat such a re-registration as a registration refresh when it is received prior to the half-life time for the specific contact. The Net-Net SBC also uses the new contact username for the Request-URI in requests it sends to the UA, and verifies that the UA uses the correct one when that Net-Net SBC is set to allow-anonymous registered mode.

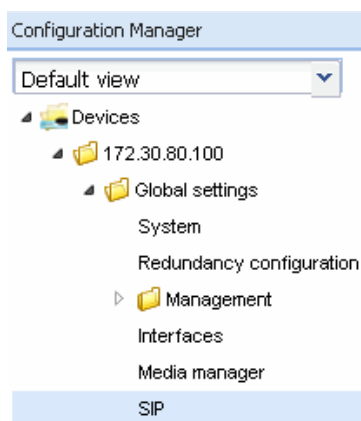
Configuring Random Registered-Contact

Enabling this feature has an impact on the registration cache mode in the global SIP configuration. The registration cache mode is set using the option **reg-cache-mode**, but regardless of how you configure it, the registration cache mode will be set to **contact** when SIP random registered-contact feature is enabled.

You enable this feature by entering a new option (**reg-contact-user-random**) in the global SIP configuration.

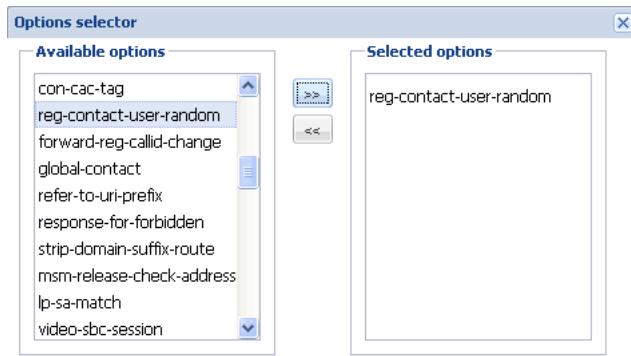
To the enable SIP random registered-contact feature:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.

- Click **reg-contact-user-random** in the Available options list to select it.



- Click  to move the option to the opposite list. The option name appears along with a description in the lower part of the window.
- Click OK.
- Click Apply.

Suppress SIP Reliable Response Support for IWF, Phase 1

For IWF-originated calls, the Net-Net SBC lets you configure the suppression of the SIP 100rel option tag on a per-H.323 interface (stack) basis.

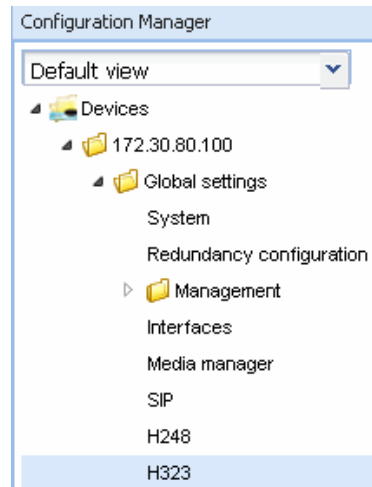
When a call originates on the H.323 side for a call that requires interworking between H.323 and SIP, the Net-Net SBC inserts the 100rel option tag in the Supported header of the outgoing SIP INVITE. Although this behavior is required for RFC 3262 conformance, and is ignored by endpoints that do not support this RFC, suppressing the reliable response can alleviate processing burdens and avoid the possibility that an endpoint could mishandle the response.

In prior releases, you can suppress the reliable response on a global basis or per SIP next-hop by using the SIP feature configuration. This supplement allows a new degree of granularity by making the functionality only applicable to IWF calls that originate in H.323.

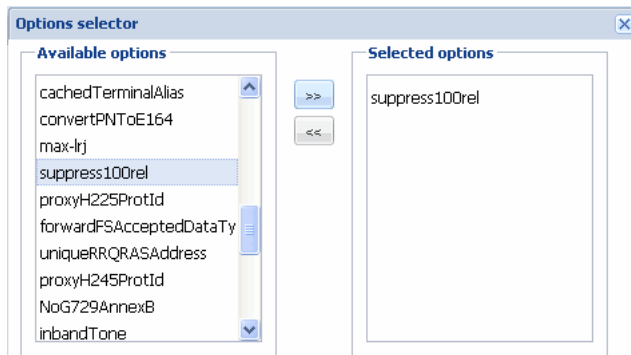
Configuring Reliable Response Support for IWF

To suppress the SIP 100rel option tag in outgoing INVITEs for H.323-originated IWF sessions:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Click **Options**.
5. Click **Supress100rel** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

Response Blocking

When you enable SIP response blocking, you can configure the Net-Net SBC to drop specified responses according to their status codes.

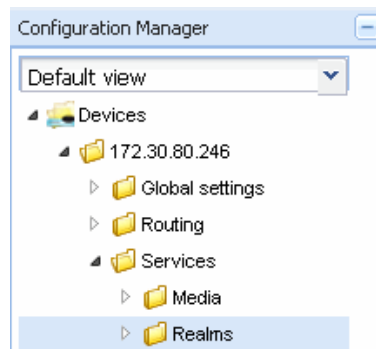
You configure this feature as an option (**dropResponse**) in the SIP interface or session agent configuration. The settings you configure in the session agent always take precedence over those in the SIP interface.

You can configure this option either with specific status codes or with status code ranges. For example, you can enter the 400 or 403 response, or you can enter 3XX to include all 300 series status codes.

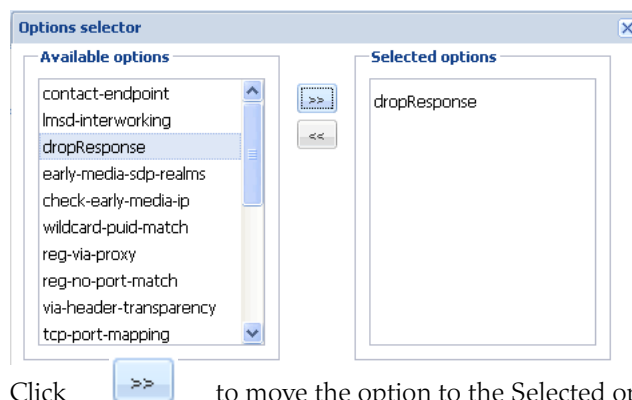
Configuring a SIP Interface

To enable SIP response blocking for a SIP interface:

1. Expand the Configuration Manager slider > Services.

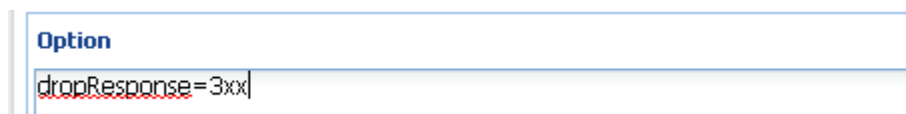


2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click **Options**.
7. Click **dropResponse** in the Available options list to select it.



8. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.

9. **dropResponse**—Edit the textbox to append specific status codes or status code ranges. For example, you can enter the 400 or 403 response, or you can enter 3XX to include all 300 series status codes.. For example:



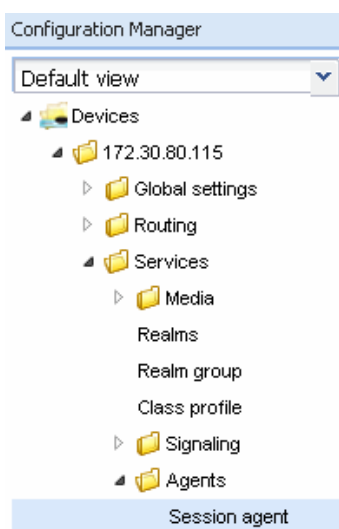
The screenshot shows a dialog box titled "Option". Inside, there is a text input field containing the text "dropResponse=3xx". A red dashed line is visible under the text in the input field.

10. Click **OK**.
11. Click **Apply**.

Configuring Session Agents

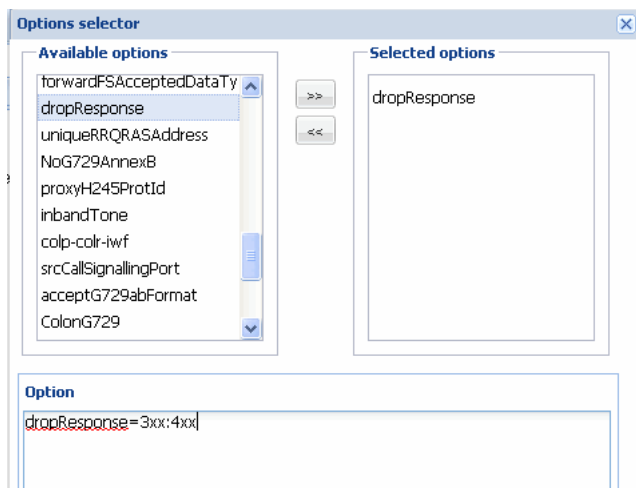
To enable SIP response blocking for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the session agent in the table and click **Edit**.
4. With the SIP session agent tab selected, scroll to the SIP options for this SA section and click **Options**. The Options dialog box appears.
5. Click **dropResponse** in the Available options list to select it.
6. Click  to move the option to the opposite list. The option name appears along with a description in the lower part of the window.

7. **dropResponse**—Edit the textbox to append specific status codes or status code ranges. For example, you can enter the 400 or 403 response, or you can enter 3XX to include all 300 series status codes.. For example:



8. Click **OK**.
9. Click **Apply**.

SDP Sendrecv Insertion

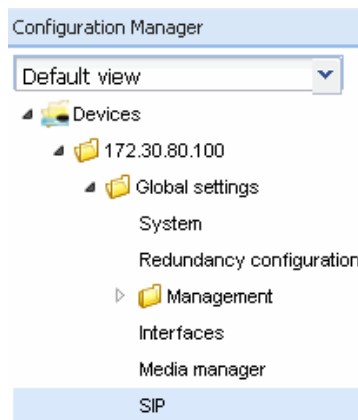
When a call is initiated, the SDP communicates between the call offerer and the call answerer to determine which route the media will take. Devices can be configured to only send media ("a=sendonly"), to only receive media ("a=recvonly"), or to do both ("a=sendrecv"). Some devices, however, do not act appropriately and do not disclose this information, which can cause problems.

In order to fix this, an enhancement has been added to the SDP. When either the offerer or answerer does not disclose its directional attribute, the Net-Net SBC automatically inserts a "sendrecv" direction attribute to the media session.

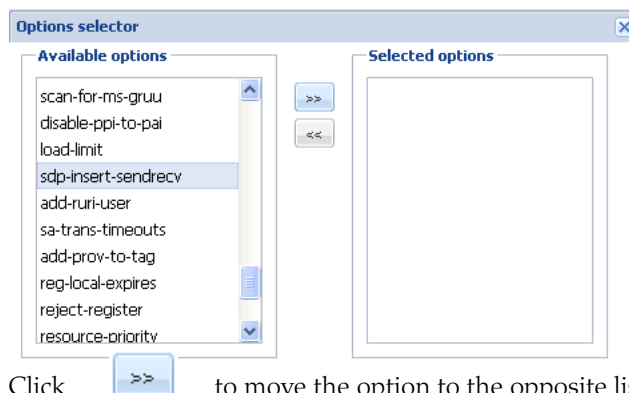
Configuring SDP sendrecv Insertion

To enable SDP sendrecv insertion:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.
5. Click **sdp-insert-sendrecv** in the Available options list to select it.



6. Click  to move the option to the opposite list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

Session Agent Status Based on SIP Response

You can configure SIP response codes to take session agents out of service or put them in service. You can set the SIP response message that either takes a session agent out of service or allows it to remain in service when it responds to the Net-Net SBC's ping request.

In prior releases, the Net-Net SBC determined session agents' health by sending them ping messages using a SIP method that you configured. Commonly, the method is an OPTIONS request. If it receives any response from the session agent, then the Net-Net SBC deems that session agent available for use.

However, issues can arise when session agents are administratively out of service, but able to respond to OPTIONS requests. A session agent like this might only respond with a 200 OK when in service, and send a 4xx or 5xx message otherwise.

Details of this feature are as follows:

- The Net-Net SBC only considers a session agent in service when it responds to a request method you set with the final response code that you also set. If a final response code is set, then provisional responses are not used for determining whether or not to take a session agent out of service. If the Net-Net SBC receives a final response code that does not match the session agent configuration, it treats the session agent as though it had not responded.
- The Net-Net SBC takes a session agent out of service when it receives an error response for dialog creating request with a response code listed in the **Response codes that put SA out of service** parameter.
- In the case where a the session agent's response has a Retry-After header, the Net-Net SBC tries to bring the session agent back into service after the period of time specified in the header. To do so, it sends another ping request.

How It Works

There are two lists you can configure in the session agent configuration to determine status:

- In-service list—Defines the response codes that keep a session agent in service when they appear in its response to the Net-Net SBC's ping request. Furthermore, the Net-Net SBC takes the session agent out of service should a response code be used that does not appear on this list.
- Out-of-service list—Defines the response codes that take a session agent out of service when they appear in its response to the Net-Net SBC's ping request.

When the Net-Net SBC receives a session agent's response to its ping request, it first checks to see if there is an in-service list of responses configured for that session agent. If the list is configured and the Net-Net SBC determines that there is a match, the session agent is deemed in service. Otherwise it takes the session agent out of service. In this way, the in-service list takes precedence over the out-of-service list. If you configure the in-service list, then the Net-Net SBC ignores the out-of-service list.

If there is no list of in-service responses for the session agent, then the Net-Net SBC checks the out of service list. If it is configured and the Net-Net SBC determines that there is a match, the Net-Net SBC removes that session agent from service. If there is no match, then the session agent is deemed in service.

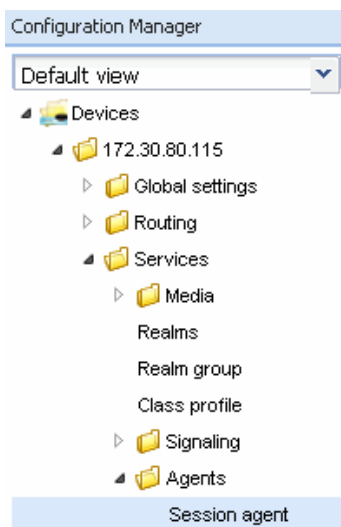
Configuring Session Agent Status

The Net-Net SBC first checks for the presence of the in-service list. If it finds one, it ignores the out-of-service list. If you want to use the out-of-service list, do not configure the in-service list. The value range is 2xx to 6xx as a comma-separated list of response codes or response code ranges. For example, 401, 404, 484-486.

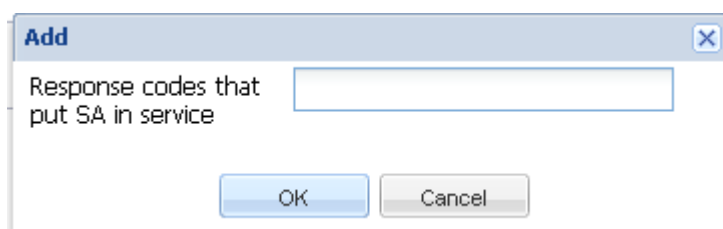
In-Service Response Codes

To configure SIP response code(s):

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a session agent and click **Edit**.
4. Click the SIP session agent constraints tab at the top of the content area.
5. **Response codes that put SA in service**—Click **Add**. The Add Inservice response code dialog box appears.
6. **Response codes that put SA in service**—Enter the response code, or range of codes, that keep a session agent in service when they appear in its response to the Net-Net SBC's ping request.

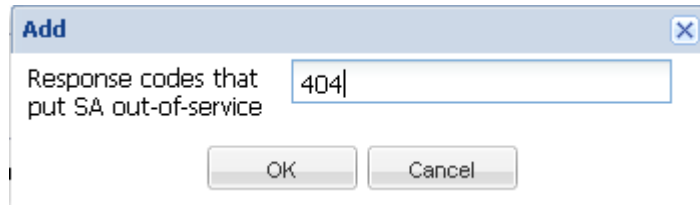


The Net-Net SBC takes the session agent out of service should a response code be used that does not appear on this list.

7. Click **OK**. The response code appears in the Response codes that put SA in service list.
8. Repeat steps 6 to 8 to enter all response codes you want to use.
9. If you do not plan to create a list of out-of-service response codes, click **Apply**.

Out-Of-Service Response Codes

1. **Response codes that put SA out-of-service**—Click **Add**. The Add Inservice response code dialog box appears.
2. **Response codes that put SA out-of-service**—Enter the response code, or range of codes, that keep a session agent out-of- service when they appear in its response to the Net-Net SBC's ping request.



3. Click **OK**. The response code appears in the Response codes that put SA out-of-service list.
4. Repeat steps 1 to 3 to enter all response codes you want to use.
5. Click **Apply**.

Session Agent Group Recursion

You can configure a SIP session agent group (SAG) to try all of its session agents rather than to the next-best local policy match if the first session agent in the SAG fails. When you enable SIP SAG recursion, the Net-Net SBC will try the additional session agents in the selected SAG if the previous session agent fails. You can also set specific response codes in the SAG configuration that terminate the recursion. This method of terminating recursion is similar to the Net-Net SBC's ability to stop recursion for SIP interfaces and session agents.

In prior releases, the Net-Net SBC performs routing by using local policies, trunk group URIs, cached services routes, and local route tables. Local policies and trunk group URIs can use SAGs to find the most appropriate next-hop session agent based on the load balancing scheme you choose for that SAG: round robin, hunt, proportional distribution, least busy, and lowest sustained rate.

When it locates a SAG and selects a specific session agent, the Net-Net SBC tries only that single session agent. Instead of trying other members of the SAG, the Net-Net SBC recurses to the local policy that is the next best match. This happens because the Net-Net SBC typically chooses a SAG based on the fact that it has not breached its constraints, but the Net-Net SBC only detects failed call attempts (due to unreachable next hops, unresolved ENUM queries, or SIP 4xx/5xx/6xx failure responses) after it has checked constraints. So the Net-Net only re-routes if there are additional matching local policies.

How It Works

Session agents are selected according to the strategy you set for the SAG, and these affect the way that the Net-Net SBC selects session agents when this feature enabled:

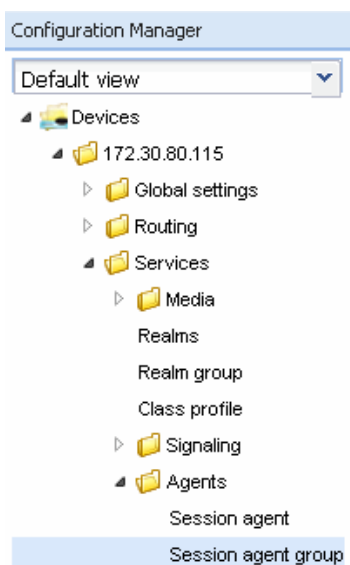
- Round robin and hunt—The Net-Net SBC selects the first session agent according to the strategy, and it selects subsequent session agents based on the order they are entered into the configuration.
- Proportional distribution, least busy, and lowest sustained rate—The Net-Net SBC selects session agents based on the list of session agents sorted by the criteria specified.

You can terminate recursion based on SIP response codes that you enter into the SAG configuration. You can configure a SAG with any SIP response code in the 3xx, 4xx, and 5xx groups. Since you can also set such a list in the session agent configuration, this list is additive to that one so that you can define additional codes for a session agent group with out having to repeat the ones set for a session agent.

Configuring Session Agent Recursion

To enable SIP SAG recursion and set response code to terminate recursion:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent group. The Session agent group table appears in the content area.
3. Select a session agent group and click **Edit**.
4. **SAG recursion**—Select enabled from the combination box to enable SIP SAG recursion.
5. **Response codes that stop recursion**—Click **Add**. The SIP response codes dialog box appears.

6. **Response codes that stop recursion**—Enter a SIP response code or range of codes, that terminate recursion within the SAG. On encountering the specified response code(s), the Net-Net SBC returns a final response to the UAC.

Response codes that stop recursion

stopRecurseion startRecursion	Add Delete

7. Click **OK**. The response code appears in the Response codes that stop recursion list.
8. Click **Apply**.

Codec Stripping and Reordering

The Net-Net SBC can strip and reorder codecs for SIP sessions. This feature builds on the Net-Net SBC's pre-existing abilities to route by codec and reorder one codec in an SDP offer. You can reorder codecs by configuring the order of multiple codecs and removing specific codecs within the media descriptions in SDP offers.

You can enable the Net-Net SBC to perform these operations on SDP offers by configuring codec policies. Codec policies are sets of rules that specify the manipulations to be performed on SDP offers and answers. They are applied on an ingress and egress basis using the realm and session agent configurations.

How It Works

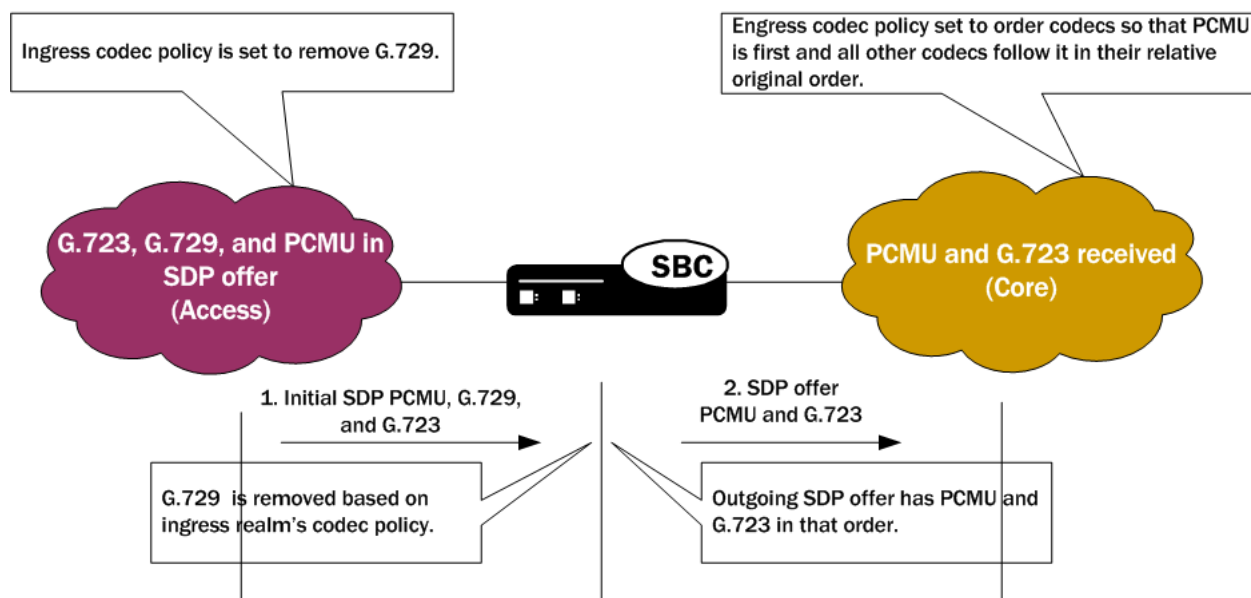
There are two types of codec policies that the Net-Net SBC applies:

- **Ingress policy**—Codec policy that the Net-Net SBC applies to the SDP offer for incoming traffic
- **Egress policy**—Codec policy that the Net-Net SBC applies to the SDP offer for traffic leaving the Net-Net SBC

The Net-Net SBC applies codec policies during the offer phase of media format negotiation. If codec manipulation is enabled, then the Net-Net SBC performs the modification according to the specific policy and forwards on the traffic.

For example, when the Net-Net SBC receives a SIP INVITE with SDP, it refers to the realm through which the INVITE arrived and performs any codec manipulations (specified in the ingress policy) that have been applied to that realm. With the media description changed according to the ingress policy, the Net-Net SBC passes it to the

outgoing realm so that the egress policy can be applied. When the second, egress policy has been applied, the Net-Net SBC forwards on the INVITE.



Because the offer-answer exchange can occur at different stages of SIP messaging, the assigned ingress and egress roles follow the media direction rather than the signaling direction. It might be, for example, that the offer is in an OK that the Net-Net SBC modifies.

You can apply codec policies to realms and to session agents; codec policies configured in session agents take precedence over those applied to realms. However, it is not required that there be both an ingress and an egress policy either for realms or for session agents. If either one is unspecified, then no modifications take place on that side. If there are neither ingress nor egress policies specified, then this feature is disabled and the Net-Net SBC will behave as it prior to this feature's introduction.

Relationship to Media Profiles

For each codec that you specify in a codec policy, there must be a corresponding media profile configuration on the Net-Net SBC. In the media profile, you can specify codec type, transport protocol, required bandwidth, and a number of constraints.

Manipulation Modes

You can configure a codec policy to perform several different kinds of manipulations:

- **Allow**—List of codecs that are allowed for a certain codec policy; if a codec does not appear on this list, then the Net-Net SBC removes it. You can wildcard this list with an asterisk (*) so that all codecs are allowed. Further, you can create exceptions to a wildcarded allow list.
- You make an exception to the wildcarded list of codecs by entering the codec(s) that are not allowed with a **no** attribute. This tells the Net-Net SBC to allow all codecs except the one(s) you specify.

- You can also create exceptions to allow lists such that audio or video codecs are removed. However, when the allow list specifies the removal of all audio codecs and an INVITE arrives at the Net-Net SBC with only audio codecs, the Net-Net SBC behaves in accordance with RFC 3264. This means that the resulting SDP will contain one attribute line, with the media port for the media line set to 0. The terminating side will need to supply new SDP in its reply because the result of the manipulation is the same as an INVITE with no body.
- Order—List of the codecs where you specify their preferred order in the outgoing media offer. The Net-Net SBC arranges matching codecs according to the rule you set, and any remaining ones are added to the list in the same relative order they took in the incoming media offer. If your list specifies a codec that is not present, then the ordering proceeds as specified but skips the missing codec.

You can use an asterisk (*) as a wildcard in this list, too. The placement of the asterisk is key, as you can see in the following examples:

- For an order rule set this way
order A B C *
 codecs A, B, and C will be placed at the front of the codec list in the order specified; all other codecs in the offer will follow A, B, and C in the same relative order they had in the original SDP offer.
- For an order rule set this way:
order * A B C
 codecs A, B, and C will be placed at the end of the codec list in the order specified; all other codecs in the offer will come before A, B, and C in the same relative order they had in the original SDP offer.
- For an order rule set this way
order A * B C
 codec A will be placed at the beginning of the codec list, to be followed by all other codecs in the offer in the same relative order they had in the original SDP offer, and then B and C will end the list.
- Force—An attribute you can use in the allow list with one codec to specify that all other codecs should be stripped from the outgoing offer. You can specify multiple forced codecs in your rules.
 - If you set multiple codecs in the allow list and one of them is forced, then the outgoing offer will contain the forced codec.
 - If you set multiple codecs in the allow list and the one that is forced is not present in the offer, then the Net-Net SBC will select a non-forced codec for the outgoing offer.**allow PCMU G729:force**
 You cannot use the force attribute with a wildcarded allow list.
- No—An attribute that allows you to strip specified codecs or codec types from a wildcarded allow list.
allow * PCMA:no

In-Realm Codec Manipulation

In addition to being able to apply codec policies in realms, the realm configuration supports a setting for determining whether codec manipulation should be applied to sessions between endpoints in the same realm.

In-realm codec manipulation can be used for simple call flows that traverse two realms. If the originating and terminating realms are the same, the Net-Net SBC checks to see if you have enabled this capability. If you have enabled it, then the Net-Net SBC performs the specified manipulations. If this capability is not enabled, or if the realm's media management in realm setting is disabled, then the Net-Net SBC does not perform codec manipulations.

For more complex calls scenarios that involve call agent or reinitiation of a call back to the same realm, the Net-Net SBC does not perform in-realm codec manipulation.

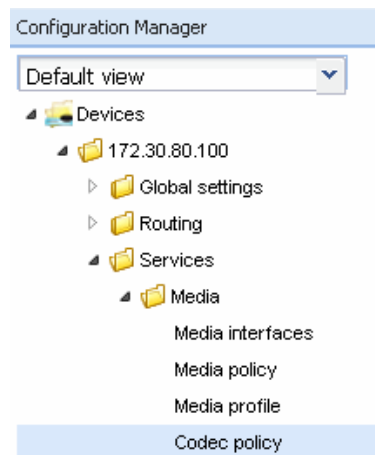
Configuring Codec Stripping and Reordering

This section gives instructions and examples for how to configure codec policies and then apply them to realms and session agents. It also shows you how to configure settings for in-realm codec manipulation.

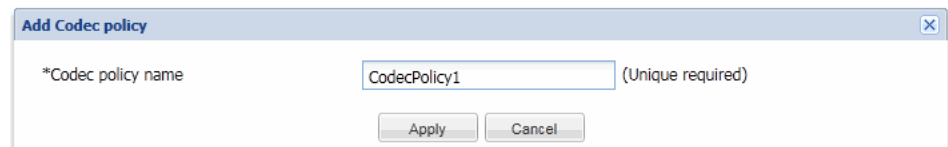
Creating a Codec Policy

To create a codec policy:

1. Expand the Configuration Manager slider > Services > Media.

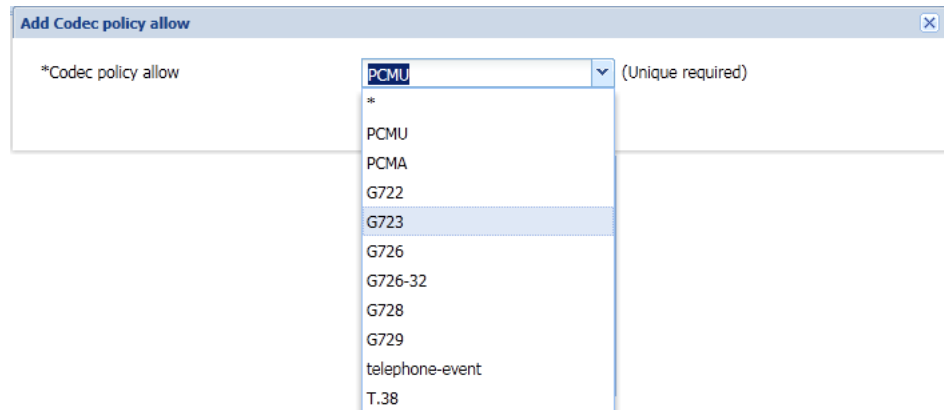


2. Click Codec policy. The Codec policy table opens in the content area.
3. Select a Codec policy and click **Edit**.
4. Click **Add**. The Add Codec policy dialog box appears.
5. **Codec policy name**—Enter the unique name for the codec policy. You refer to this codec policy name to apply it to realms or session agents. This parameter is required.



6. Click **Apply** to save the name and close the dialog box. The Codec policy name appears in the table.
7. Click the Codec in the table to select it and click **Edit**.

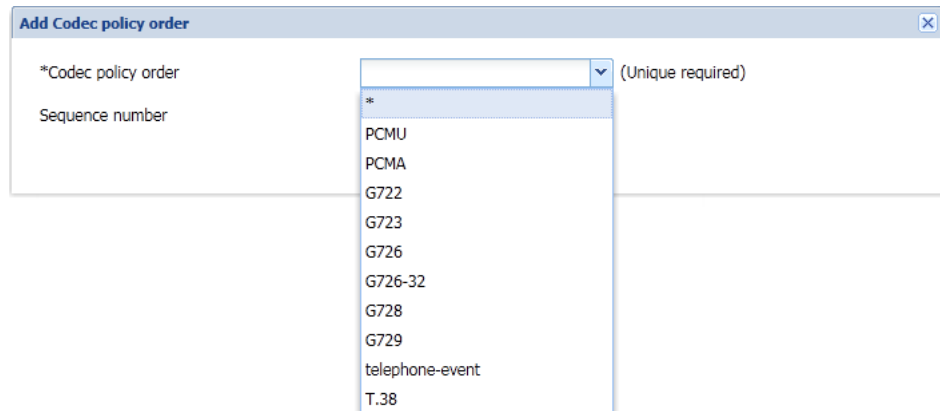
8. For Codec policy allow, click **Add**. The Add Codec policy allow dialog box appears.
9. **Codec policy allow**—Choose the media format type (codec) to allow for this codec policy from the combination box.



You can use the asterisk (*) as a wildcard, the force attribute, or the no attribute so that the allow list you enter directly reflects your configuration needs. For more information, refer to the Manipulation Modes section.

Note: The codecs that you enter here must have corresponding media profile configurations.

10. Click **Apply** to close the dialog box. The codec appears in the Codec policy allow list.
11. For Codec policy order, click **Add**. The Add Codec policy order dialog box appears.
12. **Codec policy order**—Choose the codec from the combination box.



The order in which you list the codecs is the order in which they appear in the outgoing SDP offer. You can use the asterisk (*) as a wildcard in different positions of the order to directly reflect your configuration needs. For more information, refer to the Manipulation Modes section.

Note: The codecs that you enter here must have corresponding media profile configurations.

13. Click **Apply** to close the dialog box. The codec appears in the Codec policy order list.

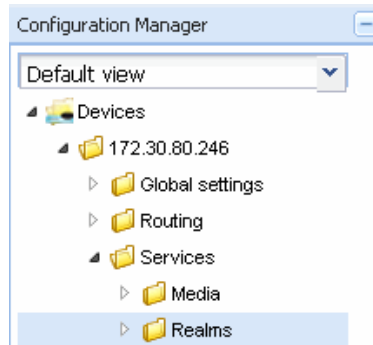
14. Click **Apply**.

Applying a Codec Policy to a Realm

Note that codec policies defined for session agents always take precedence over those defined for realms.

To apply a codec policy to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the realm table click **Edit**.
4. Scroll to Codec policy.
5. **Code policy**—Choose the name of the codec policy you want to apply to this realm from the combination box.

Codec policy

Codec policy	codec1
Codec policy in realm	enabled

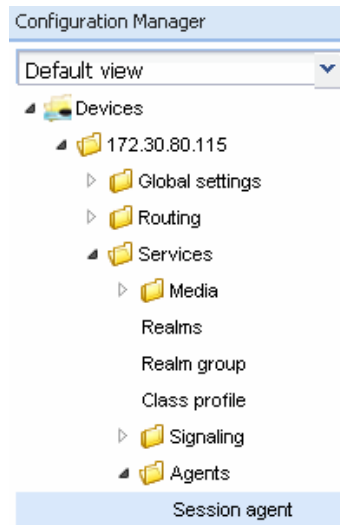
6. **Codec policy in realm**—Select enabled from the combination box.
7. Click **Apply**.

Applying a Codec Policy to a Session Agent

Note that codec policies that are defined for session agents always take precedence over those that are defined for realms.

To apply a codec policy to a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a session agent in the table click **Edit**.
4. **Code policy**— Choose the name of the codec policy you want to apply to this realm from the combination box.



5. Click **Apply**.

SIP REFER with Replaces

To support enterprise and call center applications, the Net-Net SBC provides the ability for one party participating in a three-way call to request direct connectivity between the other two parties and to leave the call silently when that connectivity is established. SIP supports this function using the Replaces header in a REFER message, also known as “REFER with Replaces.”

The most common application of REFER with Replaces handling occurs in a high-level sequence like this:

1. The customer calls a customer service line and reaches—via the Net-Net SBC—an IVR/ACD (Interactive Voice Response system/Automatic Call Distribution system). In some architectures, these are two separate elements.
2. Based on the customer’s selection from the menu of options, the IVR/ACD contacts an agent via the Net-Net SBC.
3. Since the ultimate goal is for the IVR/ACD to drop out of the path, it sends a REFER with Replaces to the Net-Net SBC. This message indicates the Net-Net SBC should replace the IVR/ACD endpoint in the call leg with the agent’s endpoint.

4. The Net-Net SBC processes the REFER with Replaces, issuing ReINVITEs to the customer with the agent's parameters.
5. The IVR/ACD drops out of the media path once the bridged call between the customer and the agent is established.

Note that direct media connectivity between endpoints must be possible in order for the REFER with Replaces to be carried out properly. For example, if both endpoints (such as the customer and agent from the example above) are behind the same firewall, direct media connectivity should be possible. However, if one endpoint is behind a firewall and the other is not, then direct media connectivity may not be possible.

For licensing capacity purposes, note that a bridged session counts as a single call.

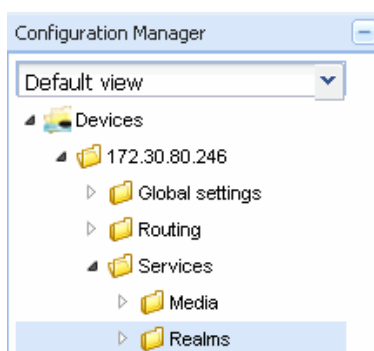
Configuring SIP REFER with Replaces

You enable SIP REFER with Replaces handling either in the realm configuration or in the session agent configuration.

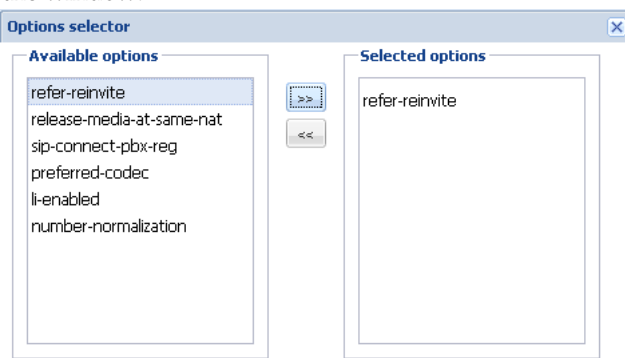
Realms

To configure SIP REFER with Replaces for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the realm table and click **Edit**.
4. Click **Options**. The Options dialog box appears.
5. Click **refer-reinvite** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.

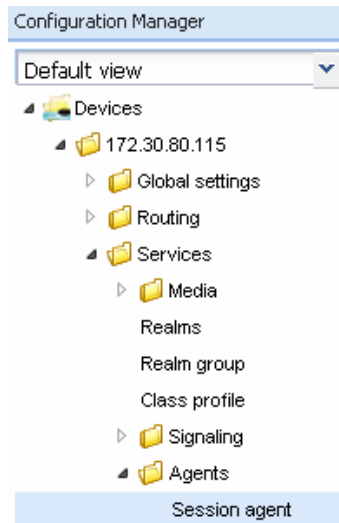


7. Click **OK**.
8. Click **Apply**.

Session Agents

To configure SIP REFER with Replaces for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the SIP session agent you want to configure in the realm table and click **Edit**.
4. Click **Options**. The Options dialog box appears.
5. Click **refer-reinvite** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

REFER Method Call Transfer

In prior releases, the Net-Net SBC supports the SIP REFER method by proxying it to the other UA in the dialog. A handling mode has been developed for the REFER method so the Net-Net SBC automatically converts a received REFER method into an INVITE method, allowing the Net-Net SBC to transfer a call without having to proxy the REFER back to the other UA.

This function can be configured for a specified SIP interface, a realm, or a session agent. When all three elements have the SIP REFER method call transfer functionality configured, the **SIP session agent** configuration takes precedence over **realms** and **SIP interface** configurations. If **SIP session agent** is not configured, and **realms** and **SIP interface** are, **realms** takes precedence.

How it Works

You can configure the Net-Net SBC to handle REFER methods as call transfers. The Net-Net SBC creates an INVITE message whenever it receives a REFER. The Net-Net SBC sends this INVITE message to the address in the Refer-To header. Included in the INVITE message is all the unmodified information contained in the REFER message. The previously negotiated codec is still used in the new INVITE message. NOTIFY and BYE messages are sent to the UA upon call transfer completion.

If a REFER method is received containing no Referred-By header, the Net-Net SBC adds one, allowing the Net-Net SBC to support all call agent screen applications.

The SIP REFER method call transfer feature also supports:

- Both unattended and attended call transfers
- Both successful and unsuccessful call transfers
- Early media from the Referred-To party to the transferee
- REFER method transfer from different sources within the destination realm
- REFER event package as defined in RFC 3515. This applies for situations where multiple REFER methods are used within a single dialog.
- Third party initiated REFER method signaling the transfer of a call by associating the REFER method to the dialogue via the REFER TargetDialog
- Referred-To party being in a different realm from the referrer or in the same realm .
- Associated latching allowing the Referred-To party to be latched to while the referee is still sending media.

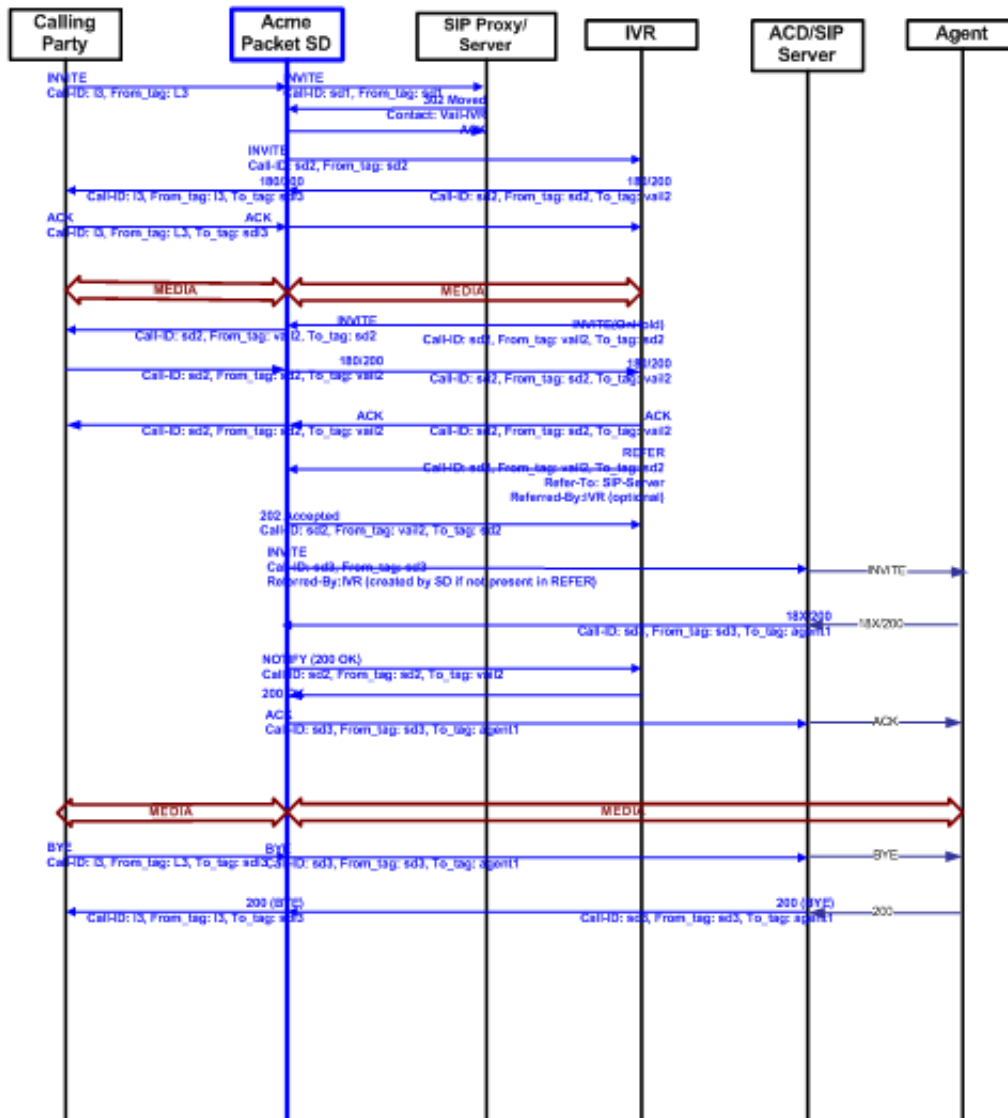
Unsuccessful Transfer Scenarios

The Net-Net SBC does not successfully handle the following failed, unusual, and unexpected transfer scenarios:

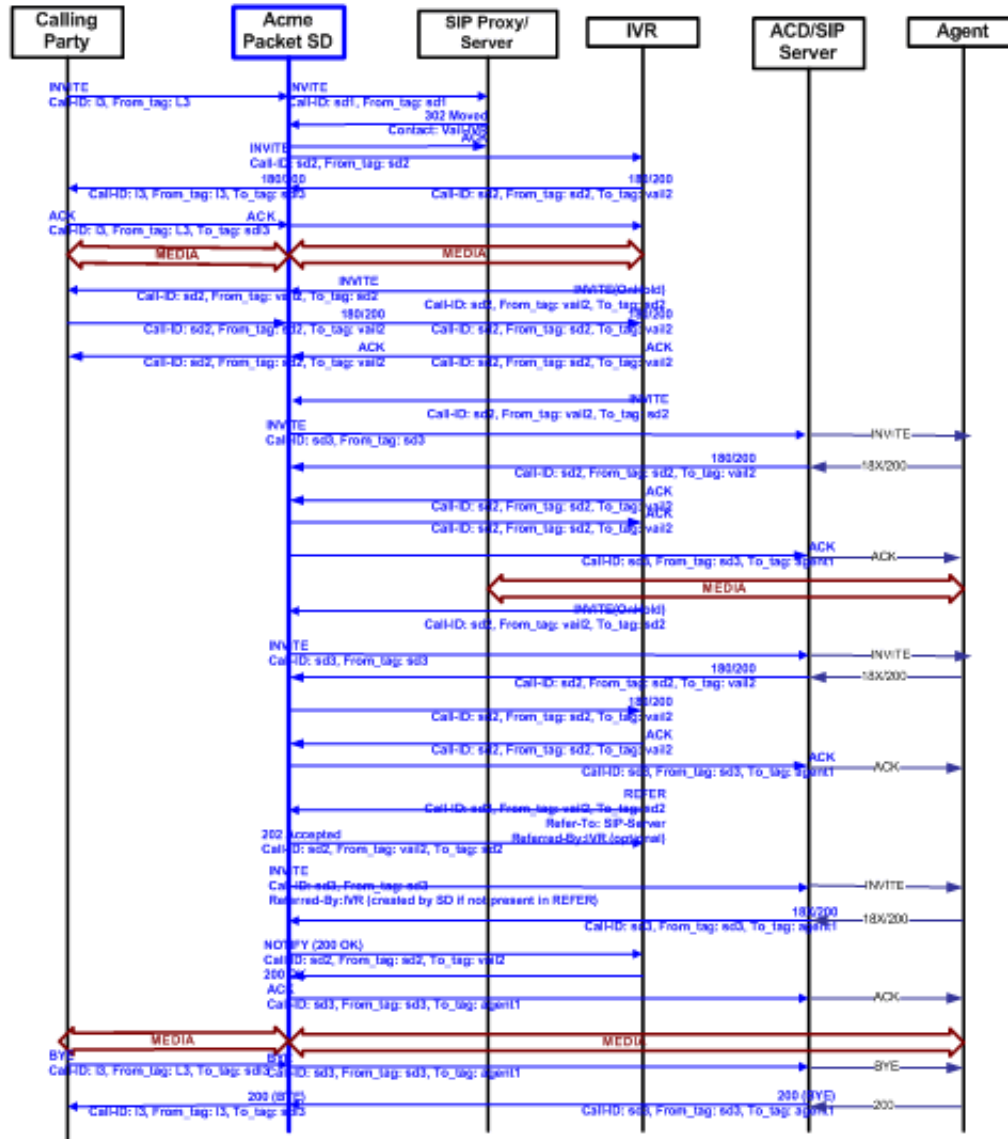
- The Net-Net SBC does not answer challenges to the new INVITE to the Referred-To party. It is treated with the 401/407 response just as any other unsuccessful final response.
- REFER message header contains a method other than INVITE or contains URI-parameters or embedded headers not supported by the Net-NET SBC.
- The Net-Net SBC allows the Referred-To URI to resolve to the same next-hop the original INVITE used.
- The Net-Net SBC ignores any REFER method containing a MIME attachment.
- The Net-Net SBC recurses (when configured to do so) when the new INVITE sent to the Referred-To party receives a 3xx response.
- The transferee indicated support for 100rel, and the original two parties agreed on using it, yet the Referred-To party does not support it.
- The original parties negotiated SRTP keys
- The original parties agreed on a codec using a dynamic payload type but the Referred-To party uses a different dynamic payload number for that codec.

Call Flows

The following is an example of a call flow for an unattended call transfer:



The following is an example call flow of an attended call transfer:



Enabling SIP REFER Method Call Transfer

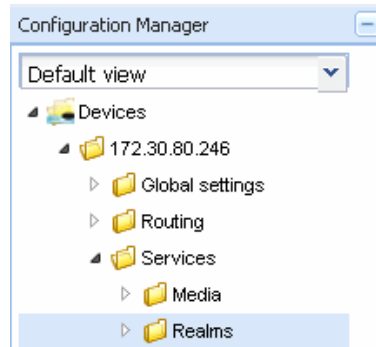
You can enable SIP REFER method call transfer for:

- realms
- SIP interfaces
- SIP session agents

Realms

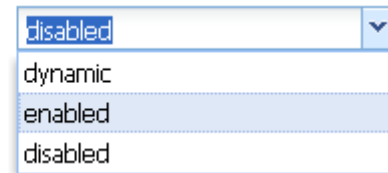
To enable SIP REFER method call transfer in realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the Realm table and click **Edit**.
4. Click the Realms tab and scroll to the **Recurse on refer call transfer** checkbox.
5. **Recurse on refer call transfer**—Select enabled from the combination box to refer call transfer feature.

Recurse on refer call transfer

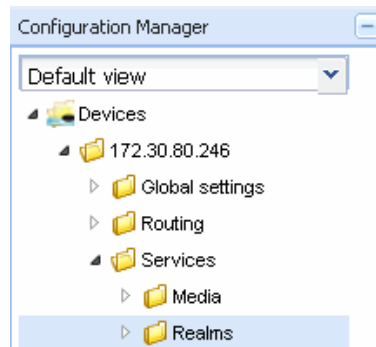


6. Click **Apply**.

SIP Interfaces

To enable SIP REFER method call transfer in the SIP interface:

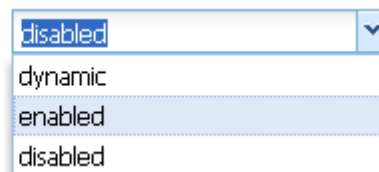
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.

3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. **Recurse on refer call transfer**—Select enabled from the combination box to refer call transfer.

Recurse on refer call transfer

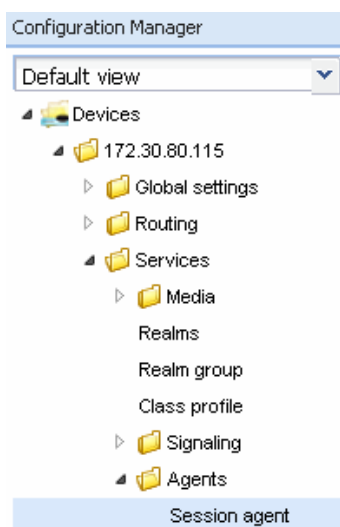


7. Click **Apply**.

SIP Session Agents

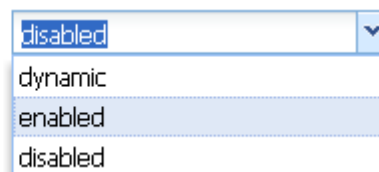
To enable SIP REFER method call transfer in the SIP session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a session agent and click **Edit**.
4. **Recurse on refer call transfer**—Select enabled from the combination box to refer call transfer.

Recurse on refer call transfer



5. Click **Apply**.

REFER-Initiated Call Transfer

In prior releases, the Net-Net SBC supported REFER-initiated call transfer either by proxying the REFER to the other User Agent in the dialog, or by terminating the

received REFER and issuing a new INVITE to the referred party. These static alternate operational modes could be configured for specific SIP interfaces, realms, or session agents.

This feature enhances support with an additional operational mode that determines on a call-by-call basis whether to proxy the REFER to the next hop, or terminate the REFER and issue an INVITE in its stead.

Note: With release of version S-C6.2.0, Net-Net SBC support for REFER-initiated call transfer is no longer available for SIP interfaces; support must be configured for realms and/or session agents.

How it Works

The configuration parameters, **dynamic refer terminated** and a revised **recurse on refer call transfer** parameter (both available in the realm configuration), specify call transfer modes.

With the **recurse on refer call transfer** parameter set to **disabled** (the default), all received REFERs are simply proxied to the peer User Agent.

With the **recurse on refer call transfer** parameter set to **enabled**, the Net-Net SBC terminates all REFERs, generates a new INVITE, and sends the INVITE to the address in the Refer-To header.

With the **recurse on refer call transfer** parameter set to **dynamic** (a new value introduced with this release), the Net-Net SBC determines REFER handling on a call-by-call basis as follows:

1. Check the **recurse on refer call transfer** value for the session agent from which the REFER was received, or for ingress realm (the realm that received the REFER).

If the value is **disabled**, proxy the REFER to the peer User Agent, to complete REFER processing.

If the value is **enabled**, terminate the REFER and issue an new INVITE to the referred party, to complete REFER processing.

If the value is **dynamic**, identify the next hop session agent or the egress realm.

2. Check the **dynamic refer terminated** value for the next hop session agent, or for the egress realm.

If the **dynamic refer terminated** value is **disabled** (the default), proxy the REFER to the next hop to complete REFER processing.

If the **dynamic refer terminated** value is **enabled**, terminate the REFER and issue an new INVITE to the referred party to complete REFER processing

Basic Scenario

In the basic scenario for REFER initiated call transfer, a call is established between two User Agents (Alice and Bob). User Agent Bob then sends a REFER request to transfer the call to a third User Agent Eva. With dynamic call-transfer enabled, the Net-Net SBC prevents the REFER from being sent to Alice and generates the INVITE to Eva.

If the INVITE to Eva succeeds, the Net-Net SBC sends a re-INVITE to Alice modifying the SIP session as described in Section 14 of RFC 3261, *SIP: Session Initiation Protocol*. At this point the Net-Net SBC cancels the original dialog between the Net-Net SBC and Bob.

If the INVITE to Eva fails, call disposition depends on whether or not Bob issued a BYE after the REFER call transfer. If the Net-Net SBC did receive a BYE from Bob (for instance, a blind transfer), it proxies the BYE to A. Otherwise, the Net-Net SBC retains the original SIP session and media session, thus allowing Bob to re-establish the call with Alice by sending a re-INVITE. In this case, the Net-Net SBC sets a timer (32 seconds), after which a BYE will be sent.

Notes

If a REFER method is received containing no Referred-By header, the Net-Net SBC adds one, allowing the Net-Net SBC to support all call agent screen applications.

In addition, the SIP REFER method call transfer feature supports the following:

- Both unattended and attended call transfers
- Both successful and unsuccessful call transfers
- Early media from the Referred-To party to the transferee
- REFER method transfer from different sources within the destination realm
- The REFER event package as defined in RFC 3515. This applies for situations where multiple REFER methods are used within a single dialog.
- Third party initiated REFER method signalling the transfer of a call by associating the REFER method to the dialogue via the REFER TargetDialog.
- The Referred-To party can be both in a different realm (and thus a different steering pool) from the referrer, and in the same realm
- The associated latching should not prohibit the Referred-To party from being latched to while the referee is still sending media.

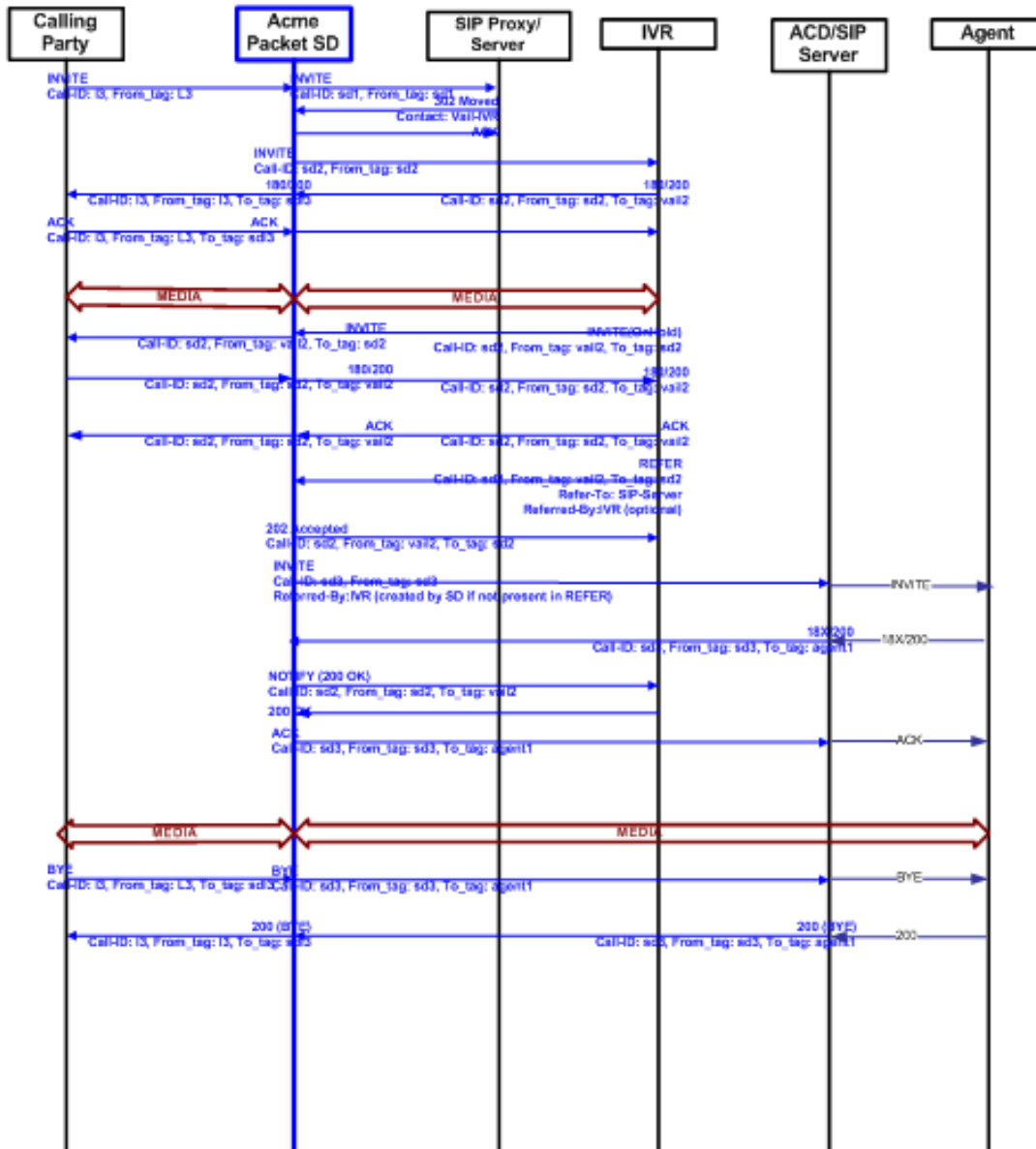
Unsuccessful Transfer Scenarios

The Net-Net SBC does not successfully handle the following anomalous transfer scenarios:

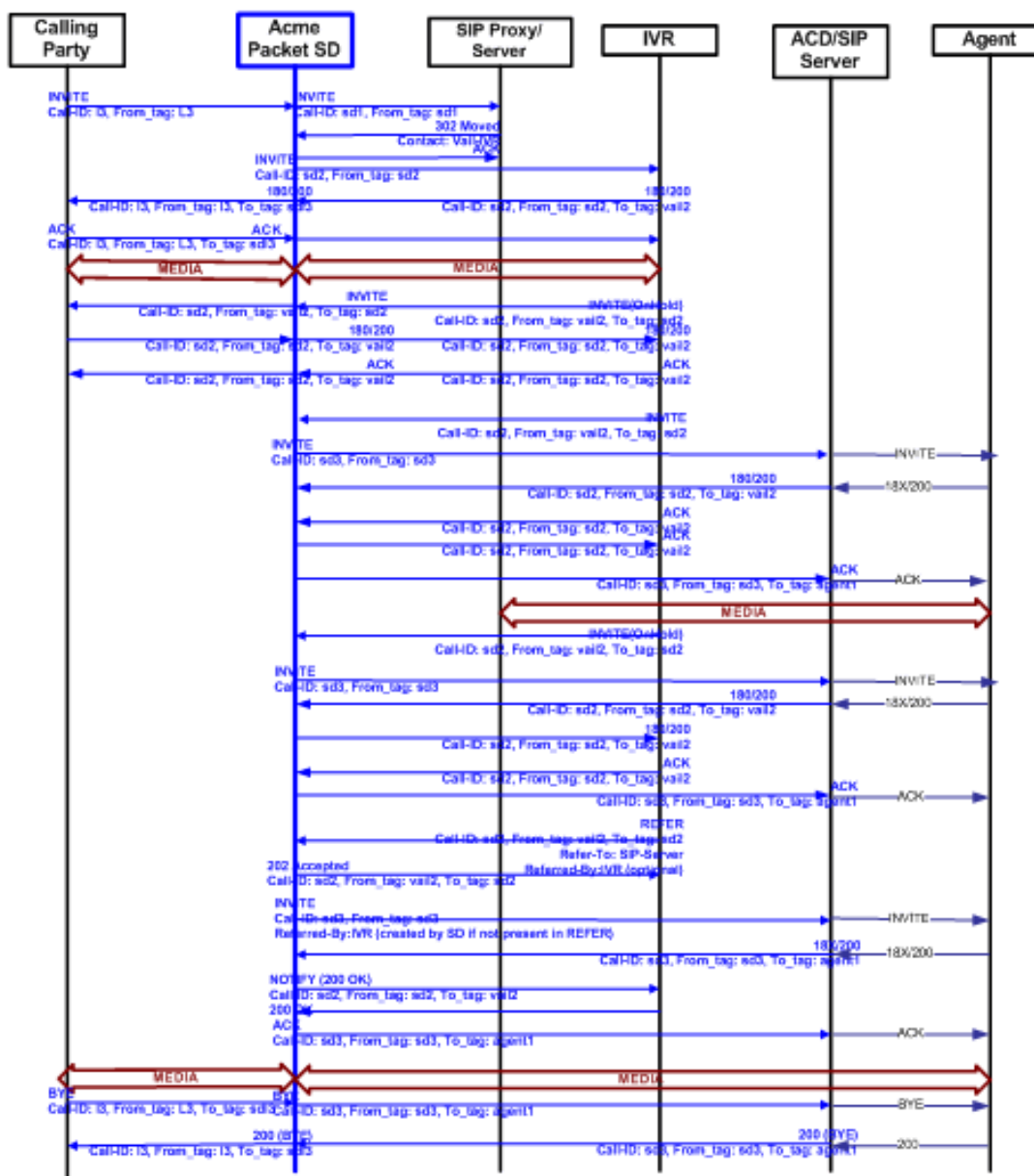
- The new INVITE to the Referred-To party gets challenged—the Net-Net SBC does not answer the challenge. It is treated with the 401/407 response just as any other unsuccessful final response.
- The header of the REFER message contains a method other than INVITE or contains URI-parameters or embedded headers not supported by the Net-Net SBC.
- The Net-Net SBC shall allow the Referred-To URI that happens to resolve to the same next-hop as the original INVITE went to, to do so.
- The Net-Net SBC ignores any REFER method containing a MIME attachment.
- The Net-Net SBC recurses (when configured to do so) when the new INVITE sent to the Referred-To party receives a 3xx response.
- The transferee indicated support for 100rel, and the original two parties agreed on using it, yet the Referred-To party does not support it.
- The original parties negotiated SRTP keys.

Call Flows

The following is an example call flow for an unattended call transfer:



The following is an example call flow of an attended call transfer:



REFER Source Routing

If, after the conclusion of static or dynamic REFER handling, the REFER is terminated and a new INVITE issued, users now can specify a policy lookup behavior based upon either the source realm of the calling party (the INVITE originator), or the source realm of the referring party (the REFER originator).

Behavior is controlled by a new **refer-src-routing** parameter in the **sip-config** configuration element.

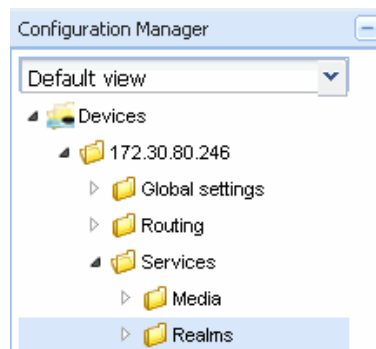
disabled, the default value, specifies that the Net Net SBC performs a policy lookup based on the source realm of the calling party.

enabled specifies that the Net Net SBC performs a policy lookup based on the source realm of the referring party.

Configuring Realm-Based REFER Method Call Transfer

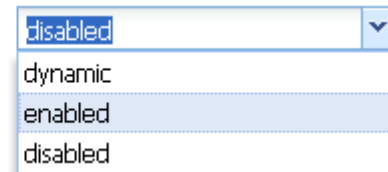
To configure realm-based REFER method call transfer:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the Realm table and click **Edit**.
4. Click the Realms tab and scroll to the **Recurse on refer call transfer** checkbox.
5. **Recurse on REFER call transfer**—Choose one of the following:
 - **disabled** (default) in the combination box to proxy all REFERS to the next hop
 - **enabled** to terminate all REFERS and issue a new INVITE
 - **dynamic** to specify REFER handling on a call-by-call basis

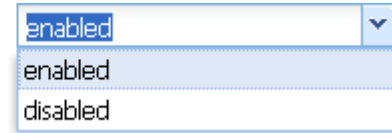
Recurse on refer call transfer



6. Click **Apply**.
7. **Dynamic REFER terminated** (meaningful only when **Recurse on REFER call transfer** is set to dynamic)—Select enabled from the combination box to proxy

the REFER to the next hop. The default state is **disabled**, which terminates the REFER and issues a new INVITE.

Dynamic Refer Termination



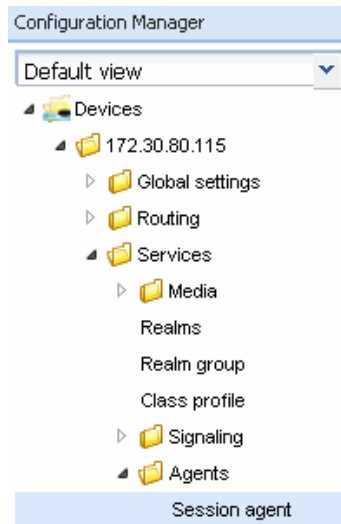
A dropdown menu with a blue border. The selected item is 'enabled'. The list of items is: enabled, enabled, disabled.

8. Click **Apply**.

Configuring Session Agent-Based REFER Method Call Transfer

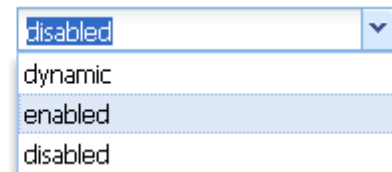
To configure session agent-based REFER method call transfer:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a session agent and click **Edit**.
4. **Recurse on REFER call transfer**—Click one of the following:
 - **disabled** (default) in the combination box to proxy all REFERs to the next hop
 - **enabled** to terminate all REFERs and issue a new INVITE
 - **dynamic** to specify REFER handling on a call-by-call basis.

Recurse on refer call transfer



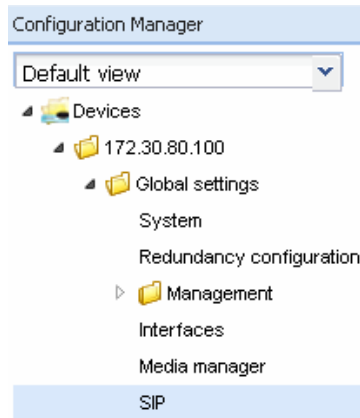
A dropdown menu with a blue border. The selected item is 'disabled'. The list of items is: disabled, dynamic, enabled, disabled.

5. Click **Apply**.

Specifying Policy Lookup

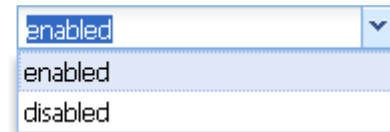
To specify policy lookup for a newly generated INVITE:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to **Refer source realm routing**.
5. **Refer source realm routing**—Select enabled from the combination box to perform a policy lookup based upon the source realm of the referring party (the issuer of the REFER). Retain the default value (**disabled**) to perform a policy lookup based upon the source realm of the calling party (the issuer of the original INVITE).

Refer source realm routing



6. Click **Apply**.

Registration Local Expiration

When you deploy multiple Net-Net SBCs with registration caching and HNT enabled in a series, registration cache entries can expire prematurely. This expiration can occur when several devices are provisioned with the same address of record (AoR). You can configure the Net-Net SBC to prevent the premature expiration.

How It Works

When you use registration caching and HNT, the Net-Net SBC adjusts the expiration time it sends to user agents (UAs) in REGISTER responses. It bases the time on the registration interval you configure.

A SIP user could have multiple registered contact endpoints at the UA to which a response is sent. If the URI in the Contact contains the UA's address and that UA included the Contact in the REGISTER request, then the Contact is seen as exclusively belonging to that UA. In the REGISTER response, this Contact (exclusive to the UA) includes the local expiration time, a time based on the SIP interface configuration's registration or NAT interval value. Additional Contacts (not exclusive to the UA) have the expiration time from the REGISTER response the registrar sent to the Net-Net SBC.

Registration cache entries might then expire prematurely in the Net-Net SBC nearest a registrar when multiple Net-Net SBCs are deployed in a series. Multiple registering UAs for a single SIP user might trigger the early expiration. You can configure the Net-Net SBC to send the local registration expiration time in all contact header Expiration parameters.

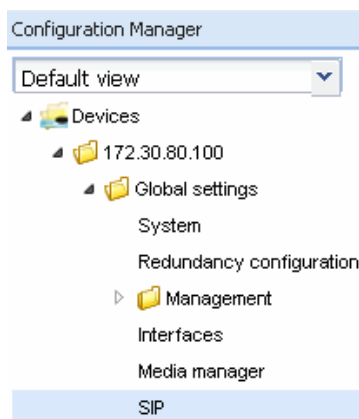
Configuring Registration Local Expiration

You can configure SIP registration local expiration for the global SIP interface configuration and for specific SIP interfaces.

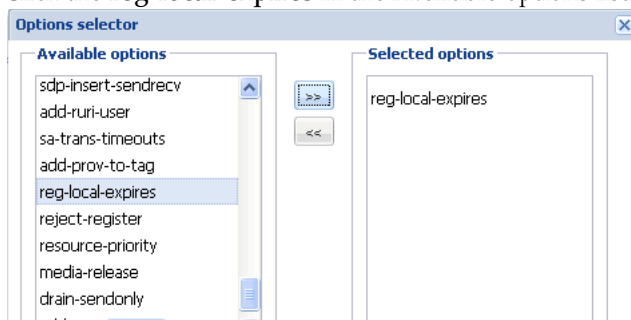
Global

To configure global SIP registration local expiration:

1. Expand the Configuration Manager slider > Global settings.



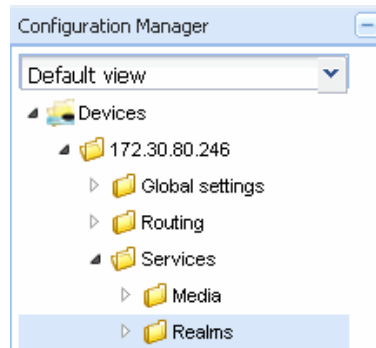
2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.
5. Click the **reg-local-expires** in the Available options list to select it.



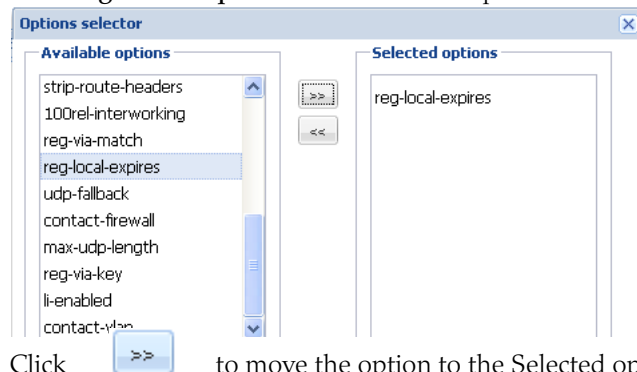
6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

Local**To configure SIP registration local expiration for the SIP interface:**

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click the Options button. The Options dialog box appears.
7. Click **reg-local-expires** in the Available options list to select it.



8. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
9. Click **OK**.
10. Click **Apply**.

Monthly Minutes-Based Call Admission Control

You can configure call admission control based on the number of minutes a month per realm to use for SIP and H.323 sessions. You configure CAC in minutes but the Net-Net SBC performs CAC to the second. When used with nested realms, the parent realm will have the total minutes for all its child realms (i.e., at least the sum of minutes configured for the child realms).

The Net-Net SBC calculates the number of minutes used when a call completes, and counts both call legs for a call that uses the same realm for ingress and egress. The total time attributed to a call is the amount of time between connection (SIP 200 OK or H.323 Connect) and disconnect (SIP BYE or H.323 Release Complete), regardless of whether media is released or not; there is no pause for calls being placed on hold.

If the number of minutes is exhausted, the Net-Net SBC rejects calls with a SIP 503 Service Unavailable message (including additional information “monthly minutes exceeded”) or an H.323 Release Complete message. In the event that the limit is reached mid-call, the Net-Net SBC continues with the call that pushed the realm over its threshold, but does not accept new calls. When the limit is exceeded, the Net-Net SBC issues an alarm and sends out a trap including the name of the realm; a trap is also sent when the alarm condition clears.

Note: The Net-Net SBC does not reject GETS/NSEP calls based on monthly minutes CAC.

You can change the value for minutes-based CAC in a realm at any time, though revising the value downward might cause limits to be reached. This value resets to zero (0) at the beginning of every month, and is checkpointed across both systems in an HA node. Because this data changes so rapidly, the value will not persist across an HA node if both systems undergo simultaneous failure or reboot.

Caveats

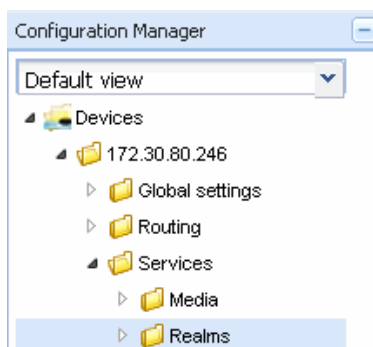
This feature is not supported for HA nodes running H.323.

Configuring Monthly Minutes-Based CAC

This section shows you how to configure minutes-based CAC for realms. Setting the monthly minutes parameters to zero (0) disables this feature.

To configure minutes-based CAC:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.

3. **Monthly minutes allowed (minutes)**—Enter the number of minutes allowed during a calendar month in this realm for SIP and H.323.calls. Range of values is 0-71582788.

Monthly minutes allowed (minutes)

2288

4. Click **Apply**.

HMR CPU Utilization Rates

The Net-Net SBC'S SIP HMR feature is improved to provide better CPU utilization rates. Some of these improvements are internal to the Net-Net SBC architecture and do not affect how you use SIP HMR. You need to be aware of other changes that have an impact on how you configure HMR.

Configuration Order

The Net-Net SBC applies SIP header rules in the order you entered them. This guards against the Net-Net SBC's removing data that might be used in the other header rules. In addition to providing this guard, the ordering provides you with ways to use manipulations strategically.

Besides taking note of the order in which header rules are configured, you must configure a given header rule prior to referencing it. For example, you must create Rule1 with the action store for the Contact header BEFORE you can create Rule2 which uses the stored value from the Contact header.

Expression Building Using Parentheses

You can use parentheses () when you use HMR to support order of operations and to simplify header manipulation rules. This means that expressions such as "(sip + urp) - (u + rp)" can now be evaluated to sip. Previously, the same expression would have evaluated to sipurprp. In addition, you previously would have been required to create several different manipulation rules to perform the same expression.

Use of Escape Characters

When you configure HMR (using SIP manipulation rules, elements rules, etc.), you can now use escape characters in the **Match value** and **New value** parameters to support escaping Boolean and string manipulation operators. .

You can also escape the escape character itself, so that it is used as a literal string. For example, the Net-Net SBC now treats the string \+1234 as +1234.

The following are escape characters: +, -, +^, -^, &, |, \, (,), ., \$, ^, and ".

New Comparison Types

The SIP header rules and the SIP element rules both contain a parameter called Comparison type, which is the way that you want SIP headers to be compared. Your new choices are:

- **boolean**—Used when you want to evaluate a referenced value by name to determine if its regular expression or Boolean expression matches TRUE or FALSE. You would choose the boolean setting when, for example, the match value is \$Rule1 or (\$Rule1 & (\$Rule2@elem4|\$Rule3.\$elem1)).

You can also use the **boolean** setting on a specific stored match to determine whether or not that match has stored a value. This is useful when you use a

regular expression that contains groupings. For example, the value `$rule1.$elem1.$2` can be used as a match value, and then if the stored match `$2` has any value, the resulting expression will be TRUE. But if the store match is empty, then the resulting expression will be FALSE.

Previously, the **pattern-rule** value included the behavior described here for **boolean**. Now, the **pattern-rule** value means only that the Net-Net SBC performs a regular expression comparison. If you are upgrading, this means that existing configurations where **pattern-rule** was set will be changed to **boolean** if the Net-Net SBC can parse the match value in the rule.

- **refer-case-sensitive**—Used when you want to perform a case-sensitive comparison of the stored value (or pattern rule subgroup) against: a header parameter value, URI parameter value, or header value.
- **refer-case-insensitive**—Used when you want to perform a case-insensitive comparison of the stored value (or pattern rule subgroup) against: a header parameter value, URI parameter value, or header value.

Performing HMR on a Specific Header

HMR has been enhanced so that you can now operate on a specific instance of a given header. The syntax you use to accomplish this is similar to that you used to refer to a specific header rule stored value instance.

Using the header-name parameter, you can now add a trailing [`<index>`] value after the header name. This [`<index>`] is a numerical value representing the specific instance of the header on which to operate. However, the Net-Net SBC takes no action if the header does not exist. You can also use the caret (^) to reference the last header of that type (if there are multiple instances)

The count for referencing is zero-based, meaning that the first instance of the header counts as 0.

Note that the header instance functionality has no impact on HMR's add action, and you cannot use this feature to insert headers into a specific location. Headers are added to the end of the list, except that Via headers are added to the top.

Multiple SIP Header Manipulation Rulesets

You can create multiple SIP header manipulation rulesets in order to:

- Apply multiple inbound and outbound manipulations rules to a SIP message
- Provision the order in which the Net-Net SBC applies manipulation rules

The Action parameter in the header rules configuration now takes the value **sip-manip**. When you set the parameter to **sip-manip**, you then configure the New value parameter with the name of a SIP manipulation rule that you want to invoke. The values for the Match value, Comparison type, and Methods parameters for invoked rules are all supported. This means that the manipulation defined by the rules identified in the New value parameter are carried out when the values for the Match value, Comparison type, and Methods parameters are true.

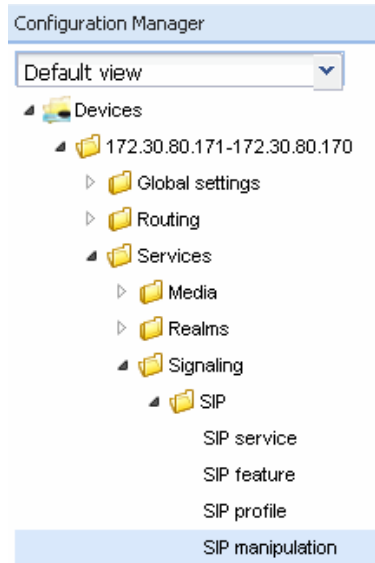
The relationship between manipulation rules and manipulation rulesets is created once you load your configuration, meaning that the order in which you enter them does not matter. It also means that the Net-Net SBC cannot dynamically perform validation as you enter rules, so you should confirm your manipulation rules contain neither invalid nor circular references. Invalid references are those that point to SIP manipulation rules that do not exist, and circular references are those that create

endless loops of manipulation rules being carried out over and over. If you load a configuration exhibiting either of these issues, the Net-Net SBC forces the Action value for the rule to **none** and the rule will not be used.

Configuring Multiple SIP HMR Sets

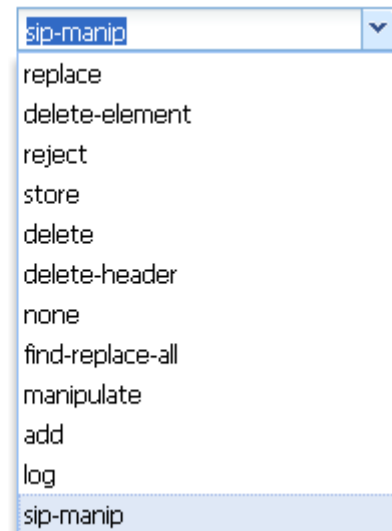
To configure multiple SIP header manipulation rulesets:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select a SIP header rule and click **Edit**.
5. Select a ruleset name in the Header, MIME and MIME ISUP rules table and click **Edit**.
6. **Action**—Choose the **sip-manip** value from the combination box. This value then invokes the rule identified in the New value parameter.

Action



7. **New value**—Enter the name of the manipulation rule you want invoked for the set.

New value

8. Click **Apply**.

Registration Burst Rate Control

The Net-Net SBC lets you control SIP registration burst rates to smooth the registration rate. The Net-Net SBC's can limit the number of:

- New register requests sent to a session agent
- Forwarded registrations from already-registered endpoints

While these methods successfully guard against the Net-Net SBC's becoming overwhelmed with register requests, registration rates can still be too aggressive and a Registrar can be overwhelmed by an excessive amount of registrations.

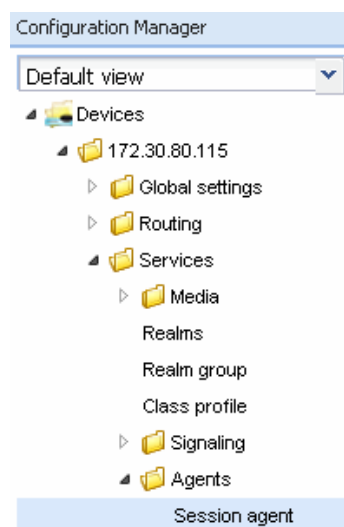
SIP registration burst rate control lets you configure SIP session agents to control the registration burst rate to limit the number of new registration requests and to set the time window for that burst rate. When the registration rate exceeds the burst rate you set, the Net-Net SBC responds to new registration requests with **503 Service Unavailable** messages.

Note: This constraint is not applied to re-registers resulting from a **401 Unauthorized** challenge request.

Configuring Registration Burst Rate Control

To configure SIP registration burst rate control:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent constraints tab.
4. Scroll to Maximum register burst rate (reg/sec) and Register burst window (sec).

5. **Maximum register burst rate (reg/sec)**—Enter the maximum number of new registrations you want this session agent to accept within the registration burst rate window. If this threshold is exceeded, the Net-Net SBC responds to new registration requests with 503 Service Unavailable messages. 0 is the minimum and default value for this parameter; the maximum value is 999999999.
6. **Register burst window (sec)**—Define the window size in seconds for the maximum number of allowable SIP registrations. 0 is the minimum and default value for this parameter; the maximum value is 999999999.

Maximum register burst rate(reg/sec)	50000
Register burst window (sec)	50000

7. Click **Apply**.

Message-Body Separator Normalization

The Net-Net SBC supports SIP with Multipurpose Internet Mail Extension (MIME) attachments. It will allow more than the required two CRLFs between the SIP message headers and the multipart body's first boundary. The first two CRLFs that appear in all SIP messages signify the end of the SIP header and the separation of the header and body of the message, respectively. At times additional extraneous CRLFs appear within the preamble before any text.

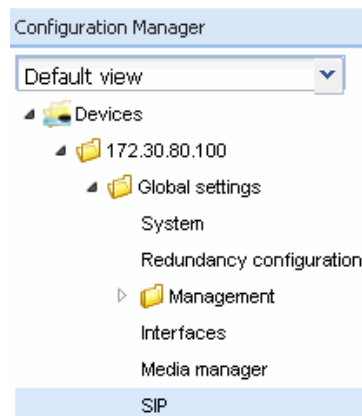
The Net-Net SBC forwards all received SIP messages, including those that contain two or more CRLFs. However, some SIP devices will not accept more than two CRLFs.

You can ensure all SIP devices accept messages sent from Net-Net SBCs by stripping all CRLFs located at the beginning of the preamble. This ensures no more than two CRLFs between the end of the last header and the beginning of the body within a SIP message.

Configuring Message-Body Separator Normalization

To configure SIP message body separator normalization:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.

3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**.
5. Click **stripPreambleCrlf** in the Available options list to select it.
6. Click  to move the option to the Selected Options list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

ENUM: Large UDP Datagram Support for DNS NAPTR Responses

The Net-Net SBC supports ENUM queries that manage large UDP DNS responses as set out in RFC 2671, “Extension Mechanisms for DNS (EDNS0).” It can manage responses beyond its previous 512 byte limitation.

In prior releases, the Net-Net SBC conforms with the DNS standard defined in RFC 1035 “Domain Names: Implementation and Specification,” which sets a maximum size for UDP responses at 512 bytes. This limitation means that responses larger than 512 bytes are truncated (set with the TC, or truncation, bit).

How It Works

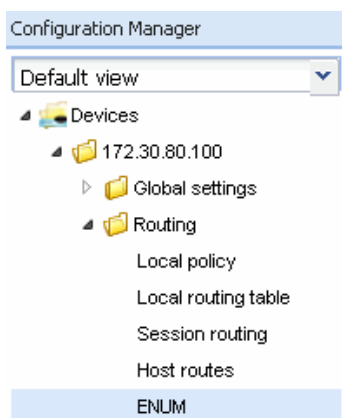
According to RFC 2671, senders can advertise their capabilities using a new resource record (OPT pseudo-RR), which contains the UDP payload size the sender can receive. You can configure the Net-Net SBC to include this resource record in its DNS queries with ENUM.

Note, however, that if your configuration does not specify a maximum response size over 512 bytes, then the Net-Net SBC will not add the OPT pseudo-RR to the ENUM query—without which the ENUM server will truncate the response.

Configuring Large UDP Datagram Support

To enable large UDP datagram support:

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.

4. **Maximum response size (bytes)**—Enter the maximum size in bytes for UDP datagram responses. The range of values is from 512 (default) to 65535. Acme Packet recommends you do not exceed 4096 bytes.

Maximum response size (bytes)

512

5. Click **Apply**.

SDP Address Correlation

SIP SDP address correlation ensures that when the Net-Net SBC receives a request containing SDP, the L3 source address of the request is compared against the address in the c-line of the SDP. When the addresses match, the session proceeds as normal. If there is a mismatch, the call is rejected with the default 488 status code. You can also configure the code you want to use instead of 488.

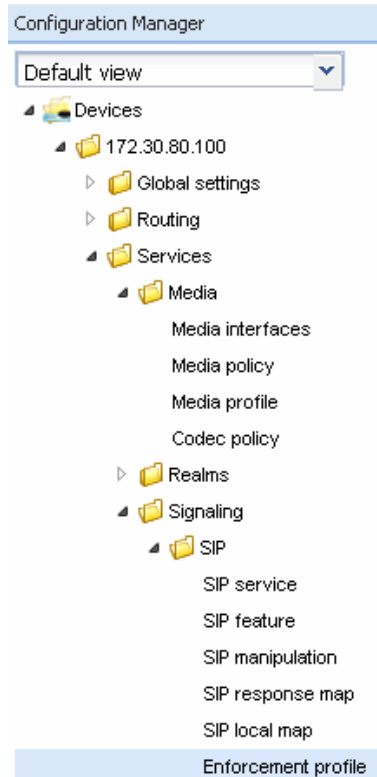
Configuring SDP Address Correlation

This functionality works only with non-HNT users. The value c=0.0.0.0 is an exception and is always processed.

Enabling SDP Address Checking

To enable SDP address checking:

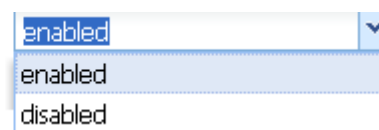
1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click **Enforcement profile**. The SIP enforcement profile table appears in the content area.
3. Select a profile and click **Edit**.

4. Select a profile and click **Edit**.
5. **SDP address check**—Select enabled from the combination box for SDP address checking.

SDP address check



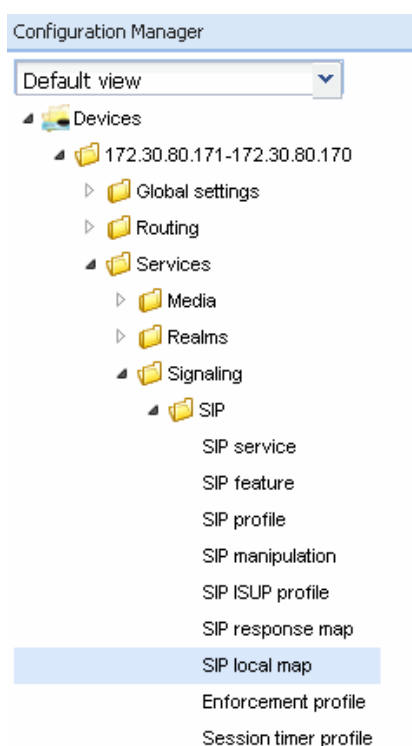
6. Click **Apply**.

If a mismatch occurs and you want to reject the call with a status code other than 488, you set the code you want to use in the local response code map entries.

Applying a New Status Code

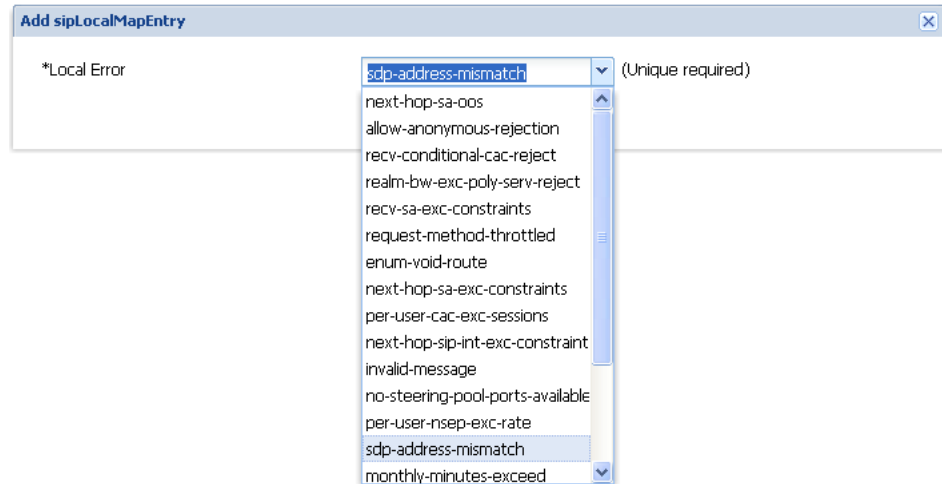
To apply a new status code to a SDP address correlation mismatch:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP local map.
3. Click **Add** to add the configuration. The SIP local map table displays in the content area.
4. Click **Add**.

5. **Local error**—Select the local error that triggers the use of this local response map from the combination box.



6. Click **Apply**.
7. Select the local map in the table and click **Edit**.
8. Select the local map from the table and click **Edit**.
9. **SIP status**—Enter the new status code you want to use, for example **403**.

SIP Status

10. **SIP reason**—Enter the reason to correlate with the new status code, for example sdp address mismatch.

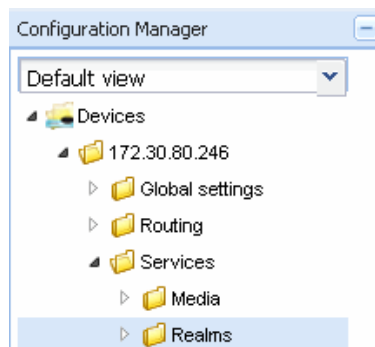
SIP Reason

11. Click **Apply**.
Apply this feature per-realm by setting the enforcement profile for a realm.

Applying the Enforcement Profile

To apply an enforcement profile to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm from the table and click **Edit**.

4. Scroll to the Enforcement profile combination box.
5. **Enforcement profile**—Choose the name of the enforcement profile you want to apply to this realm from the combination box.
6. Click **Apply**.

SDP Insertion for (Re)INVITES

You can facilitate communication between SIP endpoints that do not send SDP in ReINVITES and endpoints that refuse INVITES without SDP. The Net-Net SBC can insert SDP into outgoing INVITE messages when the corresponding, incoming INVITE does not contain SDP.

You can also use this feature when the network devices used in H.323-SIP interworking do not include SDP in the INVITES sent to SIP endpoints. In this case, the Net-Net SBC can insert SDP in the outgoing INVITE messages it forwards to the next hop.

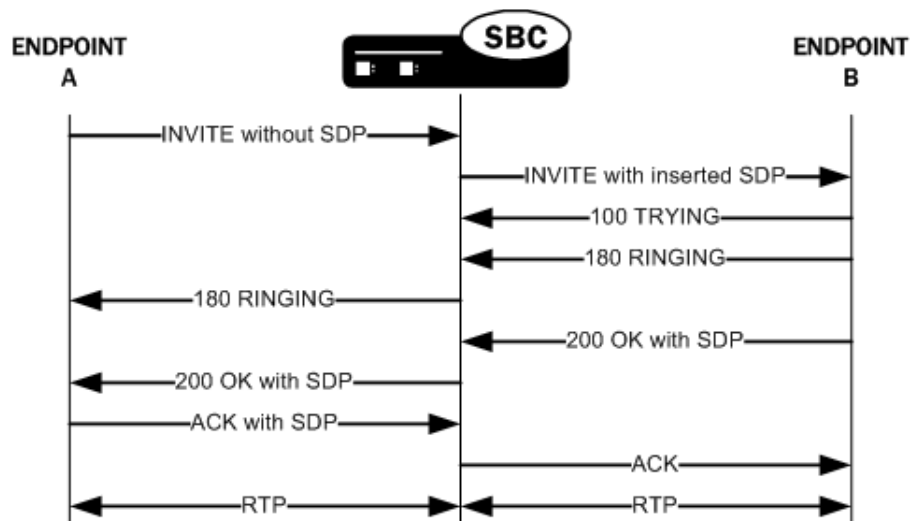
This feature works for both INVITES and ReINVITES.

How It Works

This section explains how the SDP insertion feature works for INVITES and ReINVITES. The examples used this section are both pure SIP calls. Even when you want to use this feature for IWF calls, though, you configure it for the SIP side.

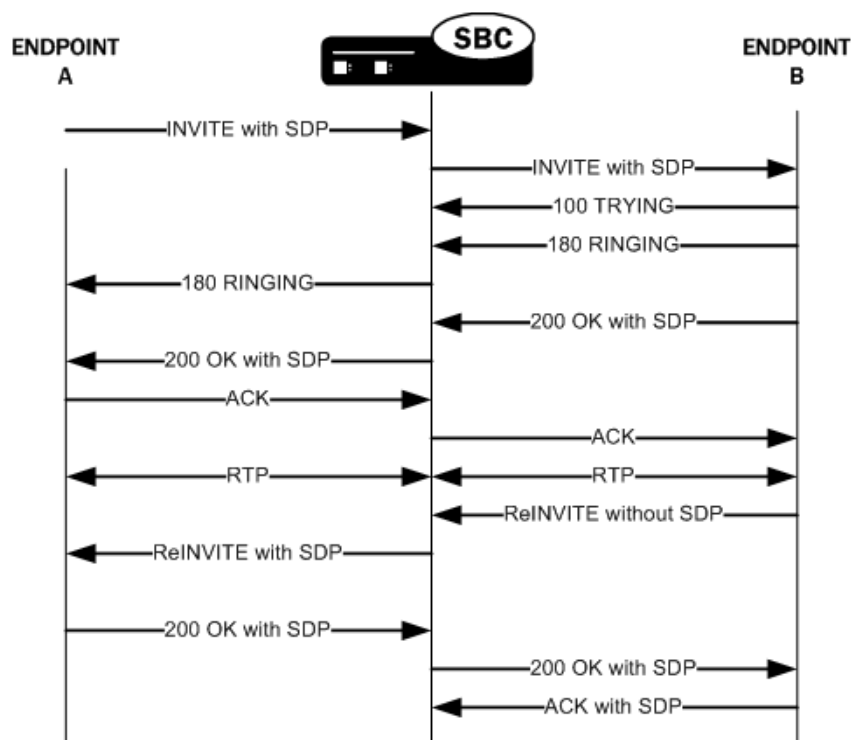
SDP Insertion for SIP INVITES

With the parameters mentioned above appropriately configured, the Net-Net SBC inserts SDP into an outgoing INVITE when the corresponding incoming INVITE has none. Because no SDP information is available for the session, the Net-Net SBC uses a media profile from a list of them you configure and then apply for SDP insertion.



SDP Insertion for SIP ReINVITES

The section explains SDP insertion for ReINVITEs, using the case where a SIP session has been established with an initial INVITE containing SDP. In the following diagram, you can see the initial INVITE results in a negotiated media stream. But after the media stream is established, Endpoint B sends a ReINVITE without SDP to the Net-Net SBC. In this case, the Net-Net SBC uses the negotiated media information from the initial INVITE to insert when the ReINVITE has no SDP. It then sends this ReINVITE with inserted SDP to the next hop signaling entity.

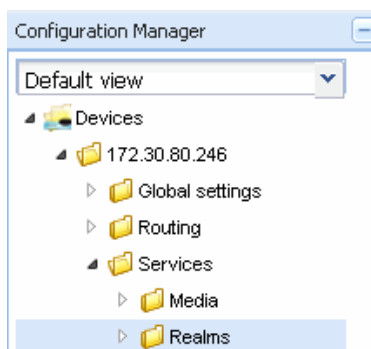


Configuring SDP Insertion for SIP INVITEs

To work properly, SDP insertion for SIP INVITEs requires you to set a valid media profile configuration. For more information about how to set up media profiles, see the *Net-Net Configuration Guide*.

To enable SDP insertion for INVITEs:

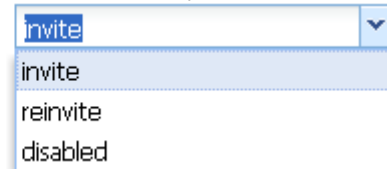
1. Expand the Configuration Manager slider > Services.



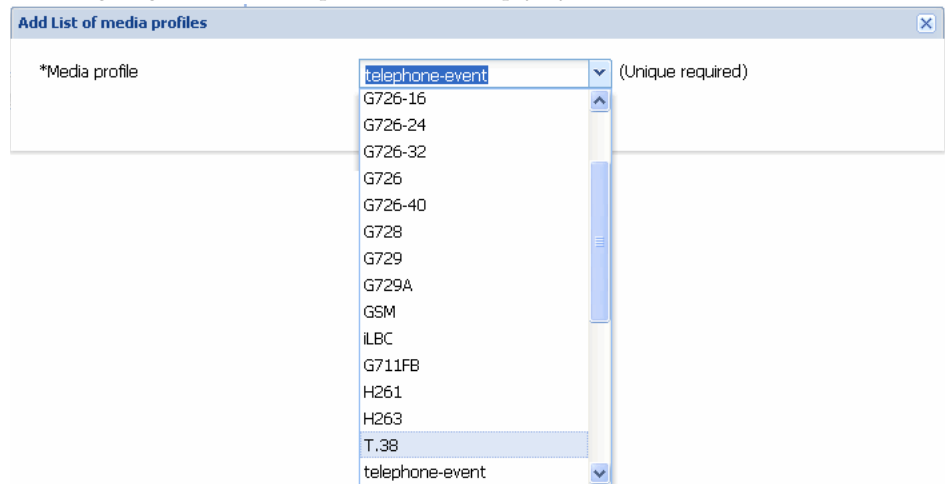
2. Click Realms. The Realm table appears in the content area.

3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to SDP Insertion re-INVITE without SDP.
7. **Insert SDP**—Click **invite** from the combination box to activate the Add option for the List of media profiles.(This parameter is disabled by default.)

Insert SDP



8. Click **Add** next to List of media profiles. The Add SIP Interface Media Profile(s) window appears.
9. **Media profile name**—Click the profile name you want to add from the combination box and click **Apply**. (These are profile name(s) you choose when the Net-Net SBC inserts SDP into incoming INVITEs that have no SDP. The media profile contains media information the Net-Net SBC inserts in the outgoing INVITE. This parameter is empty by default.).



10. Repeat steps 9 and 10 to add additional media profile names.
The List of media profiles is populated with the media profile name(s) from step 8.
11. Click **Apply**.

SIP: PRACK Interworking

When you configure your Net-Net SBC with PRACK interworking for SIP, you enable it to interwork between endpoints that support RFC 3262, *Reliability of Provisional Responses in the Session Initiation Protocol*, and those that do not.

RFC 3262 defines a reliable provisional response extension for SIP INVITEs, which is the 100rel extension tag. While some endpoints do not support the RFC, other SIP implementations require compliance with it. A session setup between two such endpoints fails. However, you can configure your Net-Net SBC to supply the

provisional response on behalf of endpoints that do not support it—and thereby enable sessions between those endpoints and the ones requiring RFC 3262 compliance.

How It Works

You need to configure PRACK interworking for a SIP interface associated with the endpoints that need RFC 3262 support. To enable the feature, you set the **100rel-interworking** option. The Net-Net SBC applies PRACK interworking for either the UAC or the UAS. The Net-Net SBC checks to see whether or not it needs to apply PRACK interworking when an INVITE arrives at the ingress or egress SIP interface with the option enabled. First, it checks the Require header for the 100rel tag; if not found there, it checks the Supported header.

Since there is a slight difference in the application of this feature between the UAC and UAS, this section explains both.

UAC-Side PRACK Interworking

The Net-Net SBC applies PRACK interworking on the UAC side when:

- A SIP INVITE contains the 100rel tag in a Require or Supported header
- The ingress SIP interface is enabled with the **100rel-interworking** option
- The UAS fails to send reliable provisional responses

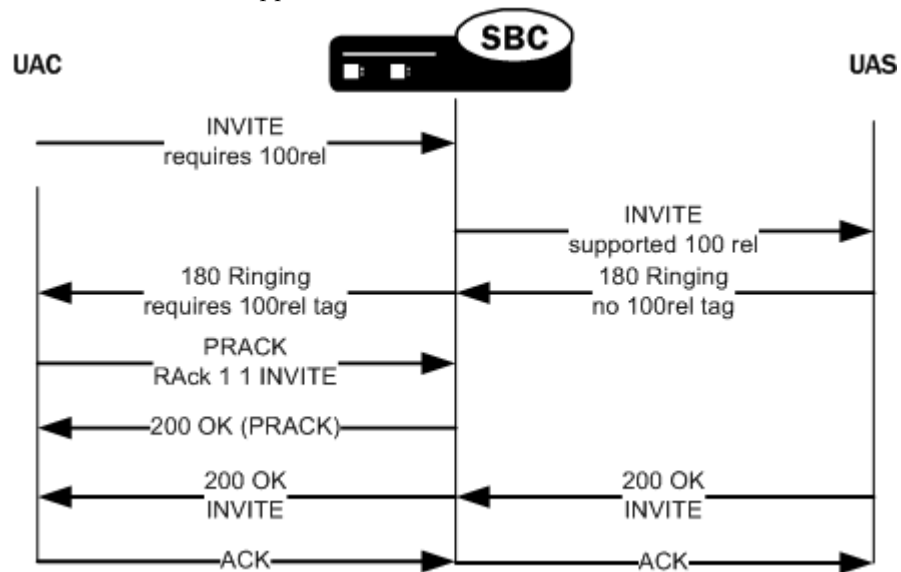
When it is to forward a non-reliable response to a UAC that requires RFC 3262 support, the Net-Net SBC converts the non-reliable response to a reliable one by adding the 100rel tag to the Require header and adding an Rseq header to the response. Further, the Net-Net SBC adds a Require header (complete with the 100rel tag) if there is not one already in the response, and then also adds Rseq header.

Note that the Net-Net SBC sets the value of the Rseq header as 1 for the first provisional response, and then increments it by 1 for each subsequent provisional response. It also adds the PRACK method to the Allow header when that header appears.

The Net-Net SBC retransmits the converted reliable provisional response in accordance with RFC 3262, until it receives a PRACK request. For the initial timeout for retransmission, the Net-Net SBC uses the value you set for the Response to INVITE initial timeout-RFC 3261 timer TI (msec) EMS label in Net-Net SBC system, SIP tab. It stops retransmitting when either it receives a transmission, or when the ingress SIP interface's trans-expire timer elapses.

If it never receives a PRACK, the Net-Net SBC does not generate an error response to the INVITE, relying instead on the downstream UAS to produce a final response.

The call flow for this application looks like this:



UAS-Side PRACK Interworking

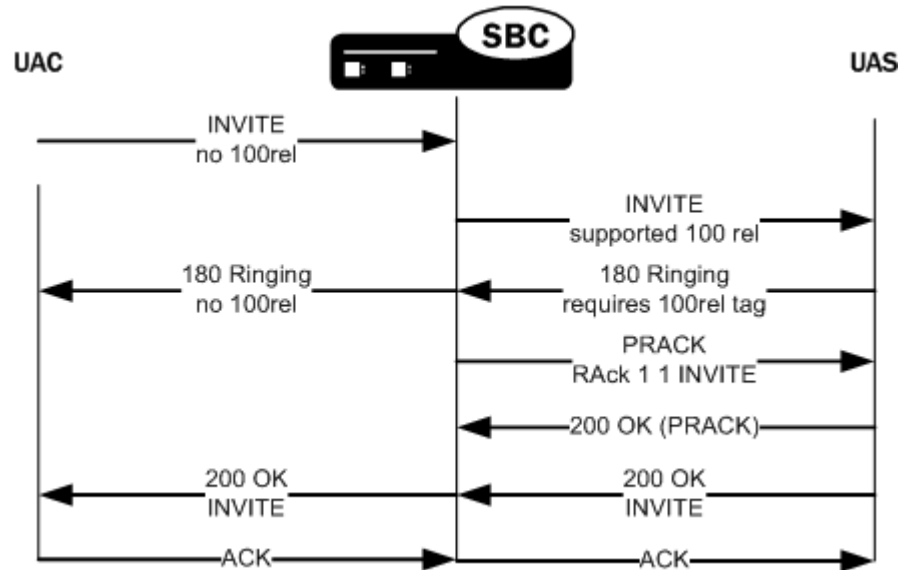
The Net-Net SBC applies PRACK interworking on the UAS side when:

- A SIP INVITE contains the 100rel tag in a Require or Supported header
- The egress SIP interface is enabled with the **100rel-interworking** option
- The UAS does send reliable provisional responses

When the UAC does not support RFC 3262, the Net-Net SBC generates a PRACK request to acknowledge the response. It also converts the response to non-reliable by removing the 100 rel tag from the Require header and removing the RSeq header from the response.

In the case of the UAS, the Net-Net SBC matches the PRACK to a converted reliable provisional response using the PRACK's RAck header. If it finds a matching response, the Net-Net SBC generates a 200 OK to the PRACK. And if it finds no match, then it generates a 481 Call Leg/Transaction Does Not Exist response. The Net-Net SBC generates a 400 Bad Request response if either the RAck is not in the PRACK request or it is not formatted properly.

The call flow for this application looks like this:

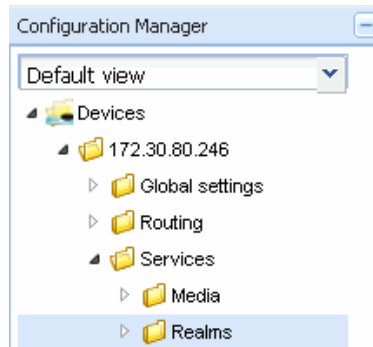


Configuring PRACK Interworking for a SIP Interface

You enable PRACK interworking for ingress and egress SIP interfaces. Be sure you know on what side, ingress or egress, you need this feature applied.

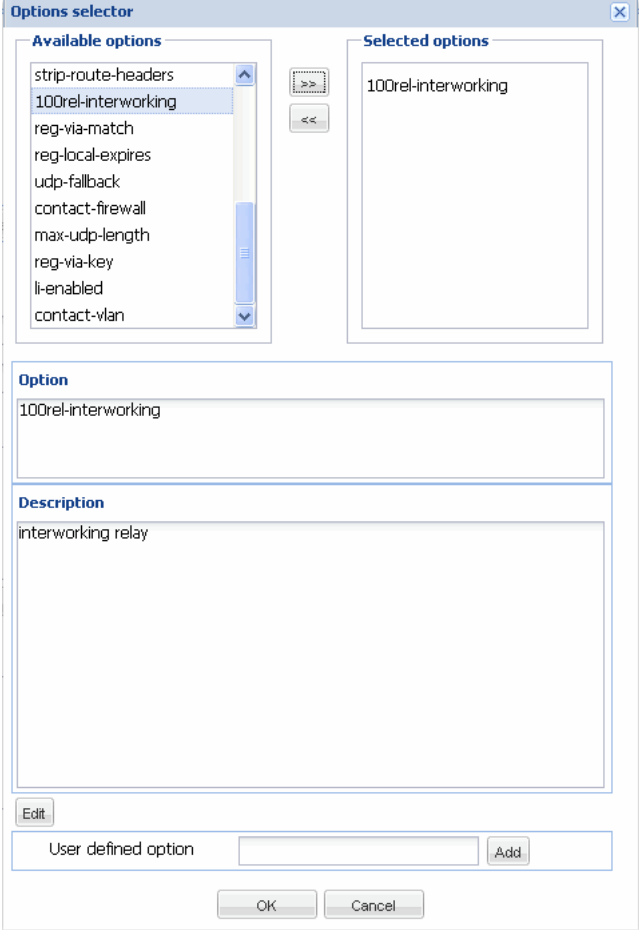
To configure PRACK interworking for a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Click the SIP interface tab.
7. Scroll to the Options in SIP service section.
8. Click **Options**. The Option window appears.
9. Click **100rel-interworking** in the Available options list.

10. Click  to move the option to the Selected options list. The option name appears along with a description in the lower portion of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. The dialog is divided into several sections:

- Available options:** A list box containing the following options: strip-route-headers, 100rel-interworking (highlighted), reg-via-match, reg-local-expires, udp-fallback, contact-firewall, max-udp-length, reg-via-key, li-enabled, and contact-vlan. There are up and down arrow buttons on the right side of the list.
- Selected options:** A list box containing the option 100rel-interworking. There are left and right arrow buttons between the Available and Selected lists.
- Option:** A text field containing the text 100rel-interworking.
- Description:** A text area containing the text interworking relay.
- Edit:** A button located below the Description field.
- User defined option:** A section with a text input field and an **Add** button.
- Buttons:** **OK** and **Cancel** buttons at the bottom of the dialog.

11. Click **OK**.
12. Click **Apply**.

NSEP Enhancements

This group of NSEP enhancements covers:

- The ability to configure and monitor NSEP call admission control (CAC) as applied to session agents
- Call treatment when the Net-Net SBC receives a SIP INVITE with an RPH matching the network management controls (NMC) with an ETS DN, but whose r-values do not match the NMC's RPH profile
- Log level changes for instances when users or session agents exceed constraints

NSEP for Session Agents

In earlier releases, the Net-Net SBC supports NSEP-related CAC for users and for NMC. You can now configure a sessions-per-second rate for session agents. Set in the global SIP configuration, this rate applies to all SIP session agents. When session exceed the limit, the Net-Net SBC rejects them with a **503 Service Unavailable** message.

Call Treatment

When the Net-Net SBC receives a SIP INVITE with an RPH matching an NMC with an ETS DN, but whose r-values do not match the NMC's rph-profile, the Net-Net SBC behaves as follows:

- If the INVITE does not have the **resource-priority** option tag and:
 - If the matching NMC is set to **PRIORITY**, the call will be treated as an NSEP call. If there is an rph-profile matching the r-value (not necessarily the one in the NMC), the Net-Net SBC uses the media-policy from that rph-profile for the call. The rph-policy from the NMC (if present) also applies to the call.
 - If the matching NMC is not set to **PRIORITY**, the Net-Net SBC will treat the call as a normal one.
- If the INVITE contains the **resource-priority** option tag, the Net-Net SBC will reject the call with the **417 Unknown Resource-Priority** message.

Log Level Changes

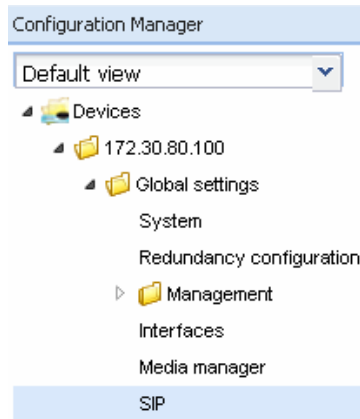
When the **NSEP user sessions rate** or **NSEP per SA sessions rate (invites per second)** parameters have been exceeded, the Net-Net SBC will log a message to **log.sipd** if the log level is as low as **NOTICE**.

Because the log level for these messages is now **NOTICE**, the Net-Net SBC must guard against instances such as an NSEP attack that would fill the log file. Therefore, only ten such log messages will be logged for a 200-second window—ten for a session agent and ten for a user, with a maximum of twenty if they fall in the same window.

Configuring NSEP Limits for Session Agents

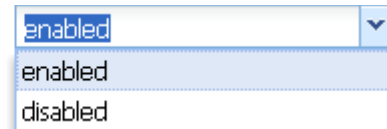
To configure NSEP limits for SIP session agents:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to Resource priority headers.
5. **RPH enabled**—Select enabled from the combination box to activate the **NSEP per SA sessions rate (invites per second)** parameter.

RPH enabled



6. **NSEP per SA sessions rate (invites per second)**—Enter the maximum acceptable number of SIP INVITES (NSEP sessions) per second to allow for SIP session agents. This parameter defaults to 0, meaning there is no limit.

NSEP per SA session rate (invites per second)

5000

7. Click **Apply**.

SIP Per User Subscribe Dialog Limit

On a per-user basis, you can limit the number of SIP SUBSCRIBE dialogs by configuring the number allowed for a given realm. Without setting these limits, the Net-Net SBC permits an unlimited number of SUBSCRIBE dialogs. Leaving the number of these dialogs uncontrolled might have potentially damaging consequences:

- Endpoints might create multiple dialogs per event package, causing undesirable Net-Net SBC resource consumption.
- If many endpoints create multiple dialogs, the problem can reach the point of creating a SUBSCRIBE DoS attack on the Net-Net SBC.

You set the restrictions this feature uses in the profiles configuration, which contains a parameter called **Event type**. Here, you enter the subscription event type and the limitations you want applied to it. You can enter more than one set of subscribe event limitations per enforcement profile. Once the enforcement profile contains the limitations you want, you then apply the enforcement profile to a realm and thereby restrict the number of SUBSCRIBE dialogs per user (as identified by the AoR).

Limit Enforcement

Enabling this feature causes the Net-Net SBC, when it receives a 200 OK or NOTIFY for a corresponding SUBSCRIBE, to determine if the dialog created by the SUBSCRIBE would exceed the configured limit for a user for this event type. When it does, the Net-Net SBC removes the oldest dialog established for the user; usually, the oldest dialog has the lowest expiration timer. However, if a user (identified by the AoR) has more than one AoC contact, then the Net-Net SBC will use the limit on a per-contact basis.

About Wildcarding

To wildcard the value, you enter an asterisk (*) for the **Event type** parameter instead of typing in the name of an actual event type.

When you wildcard this value, the Net-Net SBC applies the subscription limitations you set across all event types. Or, if you have entered multiple Event type configurations, the Net-Net SBC applies the wildcard limits across the event types for which you have not set limits.

Configuring an Enforcement Profile

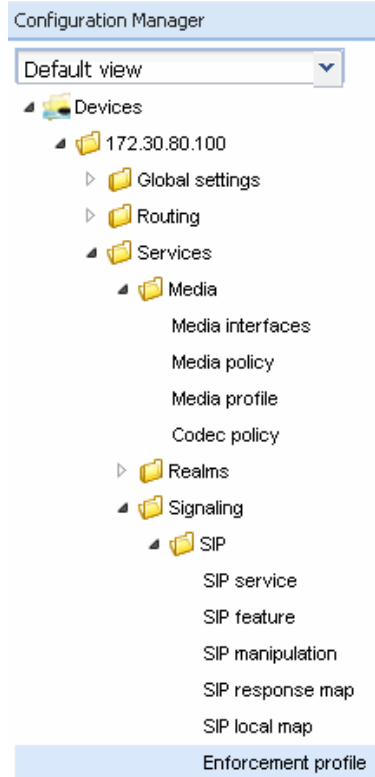
This section shows you how to configure an enforcement profile with a **Subscribe event** configuration. Remember that you can set up multiple **Subscribe event** configurations to correspond with the event types you want to control. It also shows you how to apply these limitations to a realm.

Setting Up Subscribe Dialog Limits

Setting up subscribe dialog limits means setting up an enforcement profile. For the sole purpose of setting up the subscription event limits, you only need to configure the name parameters and then as many **Subscribe event** configurations as you require. The enforcement profile has other uses, such as SIP SDP address correlation, so only configure the parameters you need.

To configure subscribe dialog limits:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click Enforcement profile. The SIP enforcement profile table appears in the content area.
3. Select a profile and click **Edit**.
4. Click **Add**. The Add Enforcement Profile window appears.
5. **Name**—Enter a name for this enforcement profile. You will use this name later when you apply the enforcement profile to a realm; it is the value you enter into the **Enforcement profile** parameter in the realm configuration.

*Name

enforcementprofile1

6. Click **OK**. The enforcement profile is added to the SIP enforcement table.
7. Click the enforcement profile and click **Edit**.
8. Click **Add** beneath the Subscribe event table. The Add Subscribe event dialog box appears.
9. **Event type**—Enter the SIP subscription event type for which you want to set up limits. The value you enter must be configured as an exact match of the event type expected in the SIP messages unless you use the wildcard. The value conforms to the event type BNF specified in RFC 3265.

You can also wildcard this value (meaning that this limit is applied to all event types except the others specifically configured in this enforcement profile). To use the wildcard, enter an asterisk (*) for the parameter value.

By default, this parameter is blank.

Add instance

enforcementProfile

ENFORCEMENTPROFILE/SUBSCRIBEEVENT

*Event type (Unique required)

10. Click **Apply**. The Event type is added to the Subscribe event table.
 11. Click the Event type and click **Edit**.
 12. **Max subscriptions**—Enter the maximum number of subscriptions allowed to a user for the SIP subscription event type you entered in the **Event type** parameter. Leaving this parameter set to 0 (default) means that there is no limit. You can set this parameter to a maximum value of 65535.
-
- *Event type
- Maximum subscriptions
13. Click **Apply**. The Max subscription is added to the Subscribe event table.
 14. Repeat steps 9 through 14 to configure multiple subscribe event types.
 15. Click **Apply**.

STUN Server

The Net-Net SBC supports RFC 3489, which defines Simple Traversal User Datagram Protocol (UDP) through Network Address Translators (NATs). Known as STUN, this is a lightweight protocol that allows applications to:

- Discover the presence and types of both NATs and firewalls between themselves and the public Internet
- Determine the public IP addresses allocated to them by the NAT

SIP endpoints use the STUN protocol to find out the public IP addresses and ports for SIP signaling and RTP media transport. Then they can use the address and port information to create multimedia sessions with other endpoints on the public network.

You can define STUN servers functionality on a per-realm basis, allowing you set up multiple STUN servers.

About STUN Messaging

STUN messages use three Binding messages and three Shared Secret messages. While it supports all three Binding messages (request, response, and error), the Net-Net SBC does not support the Shared Secret Request, or the message integrity mechanism, that relies upon the shared secret. When acting as a STUN server, the Net-Net SBC responds to STUN binding requests in accordance with RFC 3489 and the rfc3489bis draft.

STUN messages can contain the following attributes:

Message Type	Attribute Description
MAPPED-ADDRESS	Appears in the Binding Response; contains the source IP address and port from which the Binding Request was sent to the STUN server.
XOR-MAPPED-ADDRESS	Appears in the Binding Response; contains the MAPPED-ADDRESS information encoded in a way that prevents intelligent NAT devices from modifying it as the response goes through the NAT.
SOURCE-ADDRESS	Appears in the Binding Response; contains the IP address and port from which the STUN server sent its response.
CHANGED-ADDRESS	Appears in the Binding Response; contains an alternate STUN server IP address and port, different from the primary STUN server port. The STUN client might use this attribute to perform the NAT tests described in RFC 3489.
CHANGE-REQUEST	Appears in the Binding Request; instructs the STUN server to send its response from a different IP address and/or port. The STUN client might use this attribute to perform the NAT tests described in RFC 3489.
RESPONSE-ADDRESS	Appears in the Binding Request; defines an IP address and port to which the STUN server should send its responses. Appears in the Binding Request;
REFLECTED-FROM	Appears in the Binding Response; reflects the IP address and port from which a Binding Request came. Only included when the Binding Request has used the RESPONSE-ADDRESS attribute.
UNKNOWN-ATTRIBUTES	Appears in the Binding Error; reflects the mandatory attributes in a Binding Request message that the server does not support.
ERROR-CODE	Appears in the Binding Error; indicates an error was detected in the Binding Request, and contains an error code and reason phrase.

To perform NAT discovery, the endpoint (STUN client) sends a Binding Request to the STUN server port (IP address and port) with which it is configured. The STUN server then returns either a;

- Binding Response—Allows the transaction to proceed
- Binding Error—Halts the transaction, and prompts the client to take the action appropriate to the response given in the ERROR-CODE attribute

When the transaction proceeds and the STUN server sends the Binding Response, that response contains the MAPPED-ADDRESS attribute, which contains the IP address and port from which the server received the request. The STUN client then uses the MAPPED-ADDRESS when sending signaling messages.

For example, a SIP endpoint sends Binding Requests from its SIP port to determine the public address it should place in SIP headers, like the Via and Contact, of the SIP requests it sends. When this SIP endpoint prepares to make or answer a call, it sends Binding Requests from its RTP port to find out the public address it should place in SDP included in an INVITE request or response.

STUN Server Functions on the Net-Net SBC

When the Net-Net SBC receives a STUN message, it first determines its message type. Only STUN Binding Requests are processed, and all other message types are dropped without response.

Then the Net-Net SBC examines the Binding Request's STUN attributes. It returns error responses if it finds any unsupported mandatory attributes. This takes the form of a Binding Error Response, containing the ERROR-CODE attribute with reason 420 (**Unknown Attribute**) and an UNKNOWN-ATTRIBUTES attribute with a list of the unsupported attributes. If the Net-Net SBC receives a Binding Request with attributes that do not belong in STUN Binding Requests, it returns the Binding Error Response with the ERROR-CODE attribute with reason 400 (**Bad Request**).

Next the Net-Net SBC determines whether to follow RFC 3489 procedures or rfc3489bis procedures. If the Transaction ID contains the STUN cookie, then the Net-Net SBC follows rfc3489bis procedures; if not, it follows RFC 3489 procedures. Because it defines the procedures for testing the NAT to see what type of NAT it is, RFC 3489 procedures are most complex. Issues with reliability of those results have caused testing procedures and attributes to be deprecated in rfc3489bis.

RFC 3489 Procedures

The Net-Net SBC (the STUN server) constructs the Binding Response and populates it with these attributes:

- MAPPED-ADDRESS and (optionally) XOR-MAPPED-ADDRESS—Containing the source IP address and port from which the server saw the request come
- SOURCE-ADDRESS—Containing the IP address and port from which the server will send the Binding Response
- CHANGED-ADDRESS—Containing the STUN server port that has a different address and different port from the ones on which the server request was received

If the Binding Request contains a RESPONSE-ADDRESS attribute, the server adds the REFLECTED-FROM attribute with the IP address and port from which the server saw the request come. Then the server sends the Binding Response to the IP address and port in the RESPONSE-ADDRESS attribute. If the RESPONSE-ADDRESS attribute's IP address and port are invalid, the server sends a Binding Error Response with an ERROR-CODE attribute reason 400 (**Bad Request**) to the client.

If the Binding Request contains a CHANGE-REQUEST attribute, the server sends Binding Response from the IP address and port matching the information in the CHANGE-REQUEST. The following variations can occur:

- If the IP address and port flags are set, the server selects the server port with a different IP address and different port.
- If only the IP address flag is set, the server selects the server port with a different IP address but with the same port.
- If only the port flag is set, the server selects the server port with the same IP address but with a different port.

The selected server port appears in the Binding Responses's SOURCE-ADDRESS attribute. When there is no CHANGE-REQUEST attribute, the server uses the server port on which the Binding Request was received.

Finally, the server encodes the outgoing message and sends it to the client at either:

- The destination IP address and port in the RESPONSE-ADDRESS attribute, if it was present in the Binding Request.
- The MAPPED-ADDRESS.

rfc3489bis Procedures

If the Binding Request contains the appropriate cookie in its Transaction ID, the server constructs a Binding Response populated with the XOR-MAPPED-ADDRESS attribute. That attribute will contain the source IP address and port from which the server saw the request come. Then the server encodes and sends the message to the client from the IP address and port on which the request was received. The message is sent to the IP address and port from which the request came.

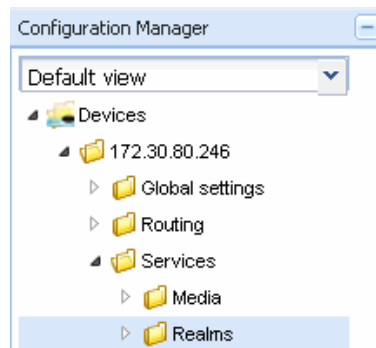
Configuring a STUN Server

You configure STUN servers on a per-realm basis, one server per realm. To support the various NAT tests it describes, RFC 3489 requires that two different IP addresses and two different UDP port numbers be used for each server. Your STUN server will listen on a total of four STUN server ports. Although newer work does away with this requirement, the Net-Net SBC supports it for the purpose of backwards compatibility.

For each realm configuration with an enabled STUN server, untrusted ACL entries will be added to forward all packets received on the four STUN Server Port.

To enable STUN server support for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm and click **Edit**.
4. Scroll to STUN.
5. **Enable server**—Select enabled from the combination box to activate STUN server support for this realm. This parameter defaults to disabled, meaning STUN server support is off.
6. **Server IP address**—Enter the IP address for the primary STUN server port. The default for this parameter is 0.0.0.0.
7. **Server IP port**—Enter the port number to use with the **Server IP address** for primary STUN server port. The default is 3478.
8. **Changed server IP address**—Enter the IP address for the CHANGED-ADDRESS attribute in Binding Requests received on the primary STUN server port. This IP address must be different from the one defined for the **Server IP address** parameter. The default for this parameter is 0.0.0.0.

9. **Changed server IP port**—Enter the port combination to define the CHANGED-ADDRESS attribute in Binding Requests received on the primary STUN server port. The default for this parameter is 3479.

STUN

Enable server	enabled
Server IP address	0.0.0.0
Server IP port	3478
Changed server IP address	0.0.0.0
Changed server IP port	3479

10. Click **Apply**.

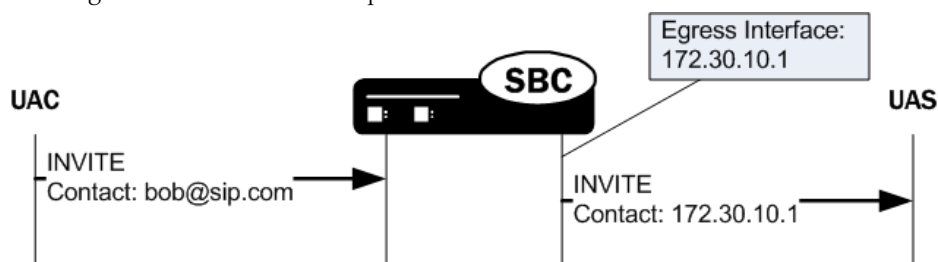
SIP Globally Routable User Agent URIs

SIP Globally Routable User Agent (UA) URIs (GRUU) are designed to reliably route a SIP message to a specific device or end user. This contrasts with a SIP AoR which can refer to multiple UAs for a single user, thus contributing to routing confusion. The Net-Net SBC can perform different behaviors when it finds SIP GRUUs in Contact headers.

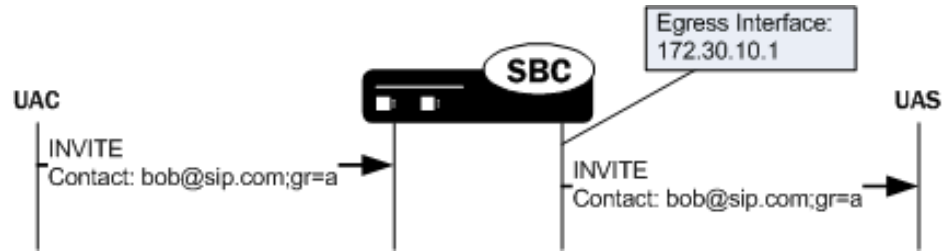
User agents supporting GRUU include a GRUU-identifying parameter in the Contact header of a dialog forming and target refresh requests. The Net-Net SBC scans for the GRUU parameter in the Contact header only when the endpoint it receives a request from is registered or when the **Pass through GRUU contact** parameter is enabled.

Contact Header URI Replacement

When no GRUU is encountered in the contact header, and when a SIP message is forwarded to the egress realm, the contact header's URI is replaced with the Net-Net SBC's egress interface. For example:

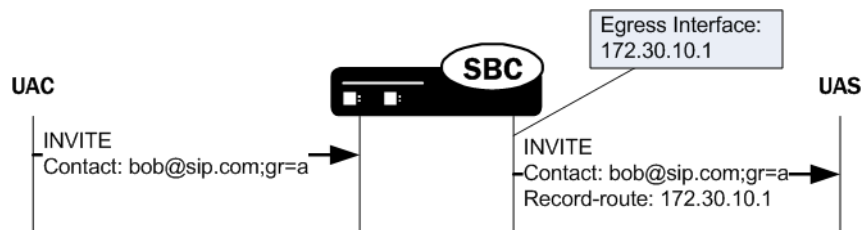


When the Net-Net SBC forwards a request where the original Contact header contains a GRUU, the contact header's URI is forwarded unchanged on the egress side of the call. For example:



Record-Route Addition

When the request is forwarded to a realm where the endpoint's registrar does not exist, the Net-Net SBC adds a Record-Route header containing the egress SIP interface address. This causes subsequent replies or requests addressed to the GRUU to be routed through the Net-Net SBC first.



When the request is forwarded to the realm where the registrar exists, adding the Record-Route header is unnecessary and does not occur. This is because subsequent requests are directed to the registrar which will ultimately forward them to the Net-Net SBC using the registered Contact in the Request-URI.

GRUU URI Parameter Name

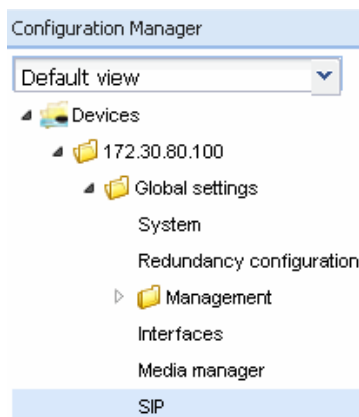
The Net-Net SBC scans for a **gr** URI parameter in the contact header to identify it as a GRUU as defined in the ietf draft[2]. The Net-Net SBC can be configured to scan for a **gruu** URI parameter in the contact header too. This alternate behavior is enabled with the **scan-for-ms-gruu** option and is used to interact with the Microsoft Office Communications Server unified communications product. When **scan-for-ms-gruu** is enabled, the Net-Net SBC scans first for the **gruu** URI parameter. If not found, it then scans for **gr** URI parameter.

Configuring SIP GRUU

This section shows you how to configure the GRUU support for non-registered contacts. Enabling GRUU functionality to parse for **gr** URI parameter rather than the IETF standard **gruu** parameter is also provided.

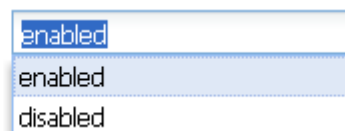
To configure SIP GRUU:

1. Expand the Configuration Manager slider > Global settings.

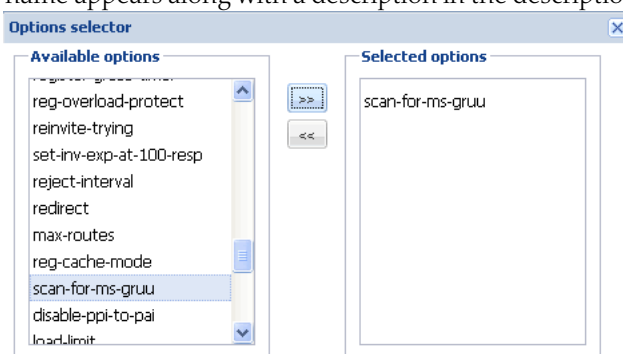


2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click the Registration tab at the top of the content area.
5. Scroll to **Pass through GRUU contact**.
6. **Pass through GRUU contact**—Select enabled from the drop-down list to activate parsing for **gr** URI parameter in the contact header in non-registered endpoints' messages and then pass the messages through the system.

Pass through GRUU contact



7. Click **Apply**.
8. Click the System tab at the top of the content area.
9. Click **Options**. The Options dialog box appears.
10. Click **scan-for-ms-gruu** in the Available options list to select it.
11. Click  to move the option to the Selected options list. The option name appears along with a description in the description section of the window.



12. Click **OK**.
13. Click **Apply**.

SIP Diversion to SIP-ISUP Interworking

For networks in which there are devices that do not support SIP-T or SIP-I (and support native SIP alone), the Net-SBC supports SIP Diversion interworking. This feature enables such devices to function properly in instances that require SIP-T/SIP-I style ISUP IAM message encapsulation in ISUP requests, and to receive any call forwarding information in the IAM according to ISUP standards.

The Net-Net SBC interworks a native SIP INVITE request to SIP-T one by inserting an ISUP IAM body based on the INVITE; this includes redirections information based on the Diversion header. This feature can also perform the reverse translation. That is, it can interwork a SIP INVITE that does have the ISUP IAM body to a non-ISUP INVITE. In this case, the Net-Net SBC generates the necessary Diversion headers based on the IAM's Redirection information.

To use this feature, you set up:

- **SIP profile**—Defines the redirection behavior
- **SIP ISUP profile**—Defines the ISUP version to use when supporting SIP-T

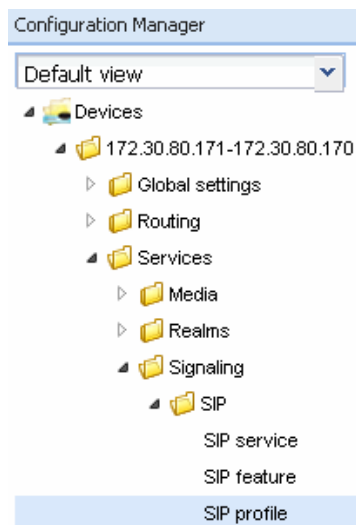
You can then apply these profiles to realms, session agents, and SIP interfaces where you want this feature enabled.

Creating a SIP Profile

The **SIP profile** configuration defines redirection behavior for the configuration where you apply it. You uniquely identify the profile so that you can apply it by name in multiple configurations. Because this is a multiple-instance configuration, you can set up as many SIP profiles as needed.

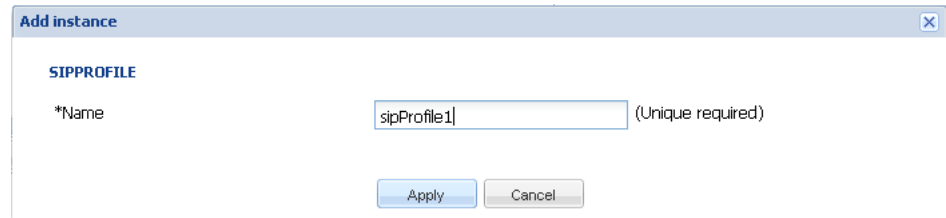
To create a SIP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.

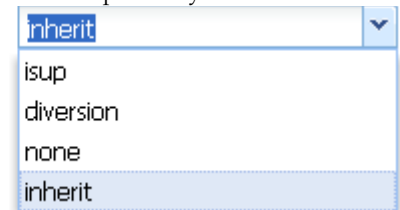
3. Click **Add**. The Add SIP profile dialog box appears.
4. **Name**—Enter the name of the SIP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.



The 'Add instance' dialog box has a title bar with a close button. Inside, the section 'SIPPROFILE' contains a label '*Name' followed by a text input field containing 'sipProfile1'. To the right of the input field is the text '(Unique required)'. At the bottom are 'Apply' and 'Cancel' buttons.

5. Click **Apply**. The new SIP profile appears in the SIP profile table.
6. Click the SIP profile you want to configure and click **Edit**.
7. **Redirection mode**—Click the redirection mode you want to use in the combination box: **none** (default), **isup**, or **redirection**. The **inherit** value is reserved for future use. When you set this parameter to **isup**, you should also configure a SIP ISUP profile to avoid future incompatibility issues.

Redirection mode



A dropdown menu showing the 'inherit' value selected. The list of options includes 'isup', 'diversion', 'none', and 'inherit'.

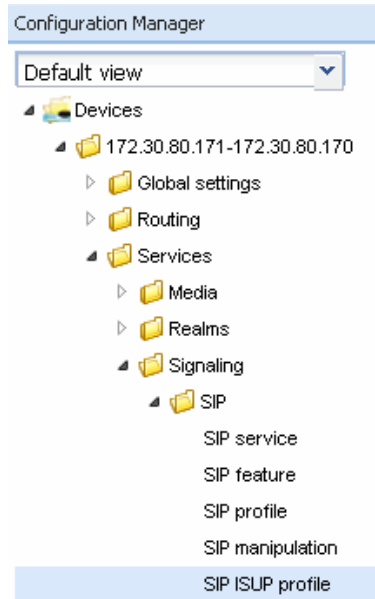
8. Click **Apply**.

Creating a SIP ISUP Profile

The **SIP ISUP profile** configuration contains information about the ISUP version you want to use. You also uniquely identify the profile so that you can apply it by name in other configurations. This is a multiple-instance configuration, meaning that you can set up as many SIP ISUP profiles as needed.

To create a SIP ISUP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP ISUP profile. The SIP ISUP profile table appears in the content area.
3. Click **Add**. The Add SIP ISUP profile dialog box appears.
4. **Name**—Enter the name for the SIP ISUP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.

5. Click **Apply**.
6. Click the SIP ISUP profile you want to configure and click **Edit**.
7. **ISUP version**—Specify the ISUP version you want used in this profile in order to support SIP-T. The default value is **ansi-2000**.

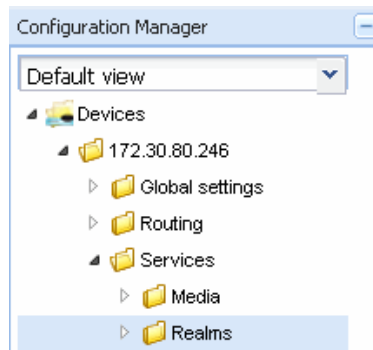
8. Click **Apply**.

Applying SIP Profiles and SIP ISUP Profiles

The following sections show you how to apply a SIP profile and a SIP ISUP profile to a realm, session agent, or SIP interface.

Realm**To apply a SIP profile and a SIP ISUP profile to a realm:**

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

SIPprofile1

4. **SIP ISUP profile**—Click the name of the SIP ISUP profile that defines the ISUP version to use for SIP Diversion SIP-ISUP interworking in the combination box. This is the value you entered in the **name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP Profile

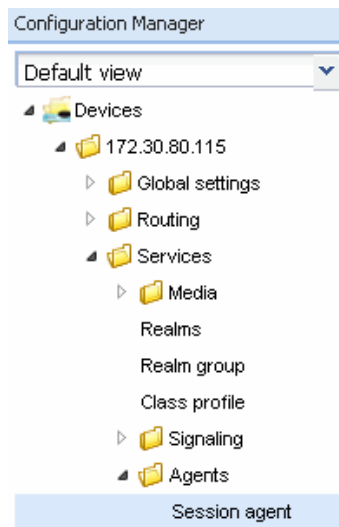
SIPisup1

5. Click **Apply**.

Session Agent

To apply a SIP profile and a SIP ISUP profile to a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

SIPprofile1

4. **SIP ISUP profile**—Click the name of the SIP ISUP profile that defines the ISUP version to use for SIP Diversion SIP-ISUP interworking. This is the value you entered in the **name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP Profile

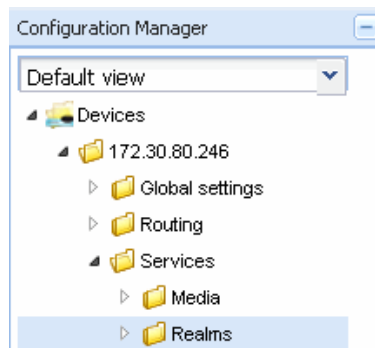
SIPisup1

5. Click **Apply**.

SIP Interface

To apply a SIP profile and a SIP ISUP profile to a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.

3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

7. **SIP ISUP profile**—Click the name of the SIP ISUP profile that defines the ISUP version to use for SIP Diversion SIP-ISUP interworking. This is the value you entered in the **name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP Profile

8. Click **Apply**.

HMR for SIP-ISUP

The Net-Net SBC's HMR functionality can operate on ISDN user party (ISUP) binary bodies. Using the same logic and mechanisms that are applied to SIP header elements, HMR for SIP-ISUP manipulates ISUP parameter fields and ISUP message parts. You can create MIME rules that function in much the same way the SIP header rules do; whereas SIP header rules can change the specific headers of a SIP message, MIME rules can manipulate targeted body parts of a SIP message.

Changes and Additions to Equality Operators

The following table defines the additions and changes to HMR equality operators introduced with this feature.

Unlike the Boolean operators the ampersand (&) and the pipe (|), you can use the following equality operators in conjunction with string operators. For example, a header value with its **comparison type** set to **boolean**, can have this match value evaluated: `"($rule1.$elem1.$0 + $rule1.$elem2.@1) == $rule2.$0"`. Equality operators can also be used with Boolean operators, as in this example: `"($rule1.$0 == $rule2.$1) & $rule3"`.

Equality operators always evaluate to either true or false.

Equality Operator Symbol	Short Description	Detailed Information
==	String case sensitive quality operator	Performs a character-by-character, case-sensitive string comparison on both the left side and the right side of the operator.
~=	String case insensitive quality operator	Performs a character-by-character, case-insensitive string comparison on both the left side and the right side of the operator.
!=	String case sensitive not equal to equality operator	Performs a character-by-character, case-sensitive string comparison on both the left side and the right side of the operator, returning true if the left side is equal to or less than the right side of the operator.

Equality Operator Symbol	Short Description	Detailed Information
>=	Greater than or equal to operator	Performs a string-to-integer conversion. If the string-to-integer comparison fails, the value is treated as 0. After the conversion, the operator will compare the two values and return true only if the left side is greater than or equal to the right side of the operator.
<	Less than operator	Performs a string-to-integer conversion. If the string-to-integer conversion fails, the value is treated as 0. After the conversion, the operator will compare the two values and return true only if the left side is less than the right side of the operator.
>	Greater than operator	Performs a string-to-integer conversion. If the string-to-integer conversion fails, the value is treated as 0. After the conversion, the operator will compare the two values and return true only if the left side is greater than the right side of the operator.

Reserved Words

To improve system performance and simplify configuration, the Net-Net SBC now supports pre-defined reserved words for commonly-used URI parameters for HMR.

Reserved words retrieve values directly from the SIP message, without your needing to create rules to store them. Their function is similar to the `$REMOTE_VIA_HOST` and other already-defined variables. If the header or value does not exist in the SIP message, either an empty string is returned or—for Boolean uses—the value `FALSE` is returned.

Reserved words apply to these commonly-accessed SIP headers and their prefixes are:

- To—`$TO_xxx`
- From—`$FROM_xxx`
- Contact—`$CONTACT_xxx`
- Request URI—`$RURI_xxx`
- P-Asserted-Identity—`$PAI_xxx`
- P-Preferred-Identity—`$PPI_xxx`
- P-Called-Party-ID—`$PCPID_xxx`

The following table contains the list of supported reserved words and a description of each.

Reserved Word	Description
<code>xxx_USER</code>	URI name of the header without any user parameters
<code>xxx_PHONE</code>	URI user of the header as a phone number but without visual separators; may or may not contains a leading plus sign (+)
<code>xxx_HOST</code>	URI host of the header
<code>xxx_PORT</code>	The URI port of the header; value set to 5060 even if it is not actually in the message

Reserved Word	Description
CALL_ID	Resolves to the Call-ID of the current SIP message; added for convenience, and is a common store rule
TIMESTAMP_UTC	Timestamp is RFC 3339 format: 2009-10-10T22:00:09Z or YYYY-MM-DDTHH:MM:SS:PPPZ. The .PPP refers to partial seconds and is optional; time is based on UTC.

The reserved word CRLF resolves to `\r\n` and is commonly used in MIME manipulation. If you are creating a new body, there might be a need for many CRLFs in the new-value parameter.

All of these operators cause additional overhead to the HMR processing because each operator requires an evaluation of the left and right sides of the expression. To speed up evaluation of new-value expressions, you can now enter escapable characters (`\f`, `\n`, `\r`, `\t`, `\v`) with a backslash (`\`) and the Net-Net SBC will convert them to escaped characters during the compilation of the expression (at ACLI configuration time).

Setting the Action Parameter

In releases prior to S-C6.2.0, the **SIP manip** action is only supported in the header. You can now set the action parameter to **SIP manip** at all levels of HMR configuration, including element rules.

About MIME Rules

MIME rules operate much the same way that SIP header rules do. You can set parameters in the MIME rules that the Net-Net SBC uses to match against specific SIP methods and message types. The system compares the search criteria against the body or body parts using the type of comparison you choose. Offering a variety of selection, you can pick the kind of manipulation that suits your needs; the Net-Net SBC then takes action with matching and new values to change the SIP message. When you use the **delete** action on a multi-part MIME string that reduces a number of bodies down to one, the SIP message remains a multi-part MIME message with only one body part (and thereby avoids the header conflicting with the message itself).

You identify the MIME rule by configuring a content type that refers to the specific body part on which to operate. For example, given a SIP Content-Type header with the value `multipart/mixed;boundary=unique-boundary-1`, you would enter a content-type value of **application/sdp** to manipulate specifically on the SDP portion of the SIP message. The Net-Net SBC knows automatically if it is operating on SIP messages with single or multiple body parts, and the content-type setting applies to both kinds. And when making its comparison, the Net-Net SBC matches the content-type of the body with regard to case (case insensitive), ignoring any header parameters.

Both for making comparisons against the body part and for new/replacement values, the Net-Net SBC treats the match and new values you set for a MIME rule as ASCII strings. Therefore, a MIME rule operating on a binary body part will yield an improper conversion of a new value with respect to the binary body part.

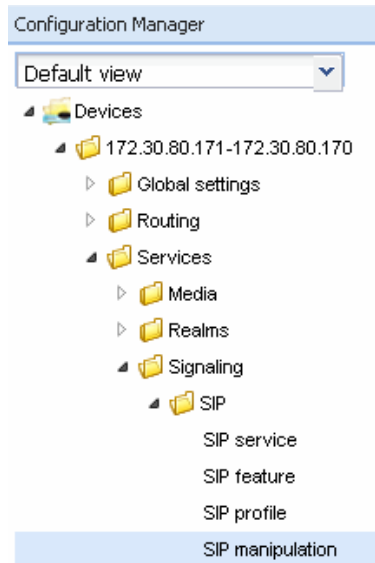
Within MIME rules, you configure MIME headers, which operate on the specific headers in the match body part of the SIP message. The Net-Net SBC uses the MIME header name to run a string comparison to match the specific header in the message's body part.

Using these rules, you can also manipulate the preamble—or the SIP message text that follows the headers but precedes the body separator. To do so, enter the keyword **@preamble** for the content type parameter in the MIME rule. Likewise you can manipulate the epilogue—or the text that follows the last body part after the last separator—using the keyword **@epilogue**.

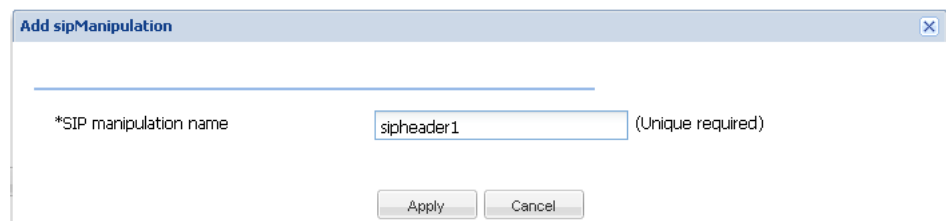
Creating MIME Rules

To create MIME rules:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.

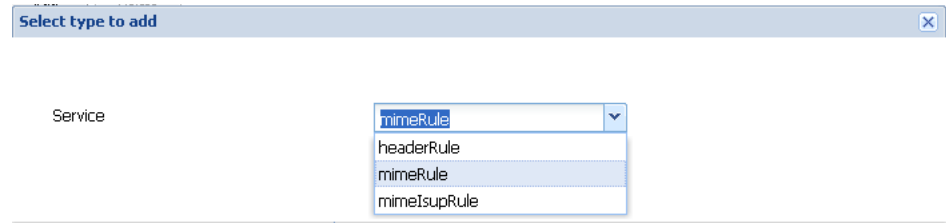


3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Click **Add**. The Add SIP manipulation rules dialog box appears.
5. **SIP manipulation name**—Type a SIP manipulation rule name as a unique identifier for this instance.



6. Click **Apply**.
7. Select the SIP manipulation rule from the table and click **Edit**.
8. Scroll to the Rules element table.
9. Click **Add**. The Select type to add dialog box appears.

10. Select mimeRule from combination box and click **Apply**.

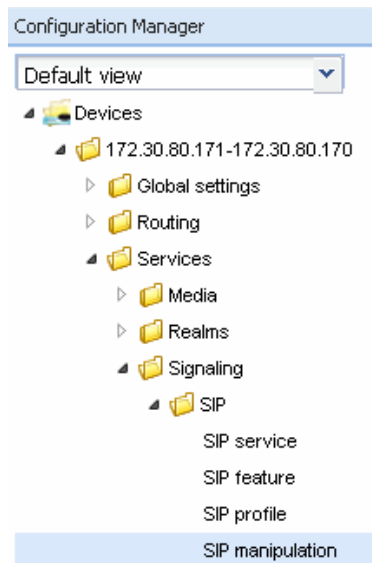


11. Click **Apply**.

Configuring MIME Rules

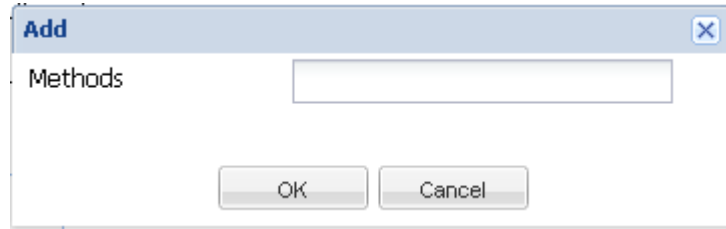
To configure MIME rules:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



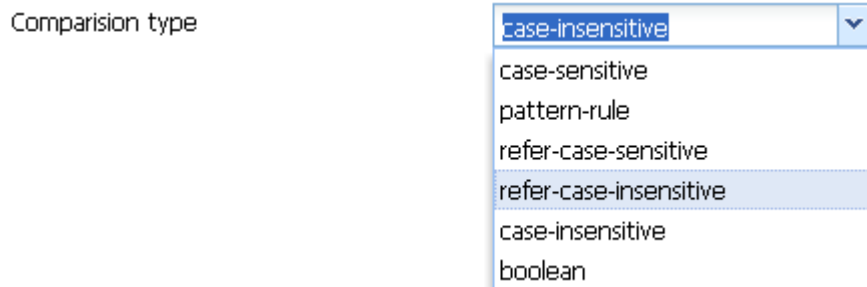
3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select the SIP manipulation rule from the table and click **Edit**.
5. Scroll to the Rules element table.
6. Select the MIME rule from the table and click **Edit**.
7. **Methods**—Click **Add**. The Add Methods dialog box appears.

8. Enter the SIP method to which the MIME rules applies. There is no default for this parameter.



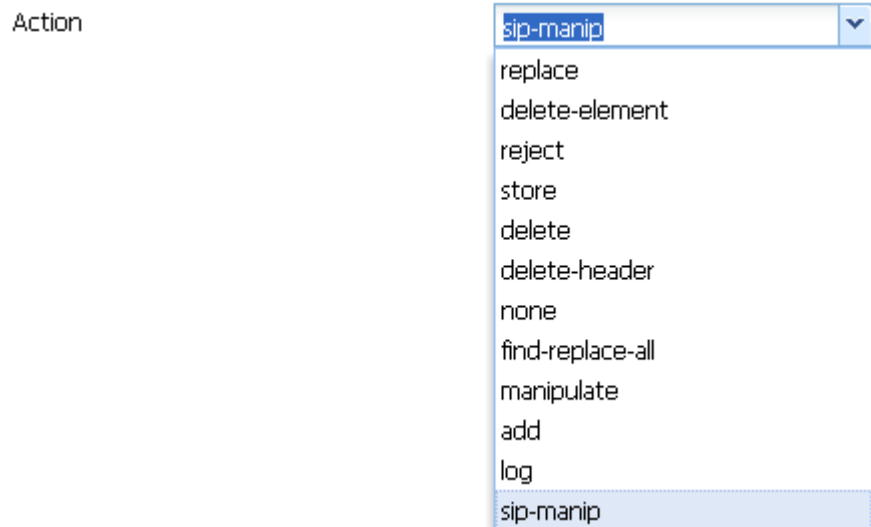
The 'Add' dialog box has a title bar with 'Add' and a close button. It contains a label 'Methods' next to an empty text input field. At the bottom are 'OK' and 'Cancel' buttons.

9. Click OK. The method is added to the Methods list.
10. Repeat steps 8 through 10 to add additional SIP methods.
11. **Comparison type**—Click the **comparison type** value in the combination box. This is the way that you want the body part of the SIP message to be compared. This choice dictates how the Net-Net SBC processes the match rules against the SIP header. The default is **case-sensitive**.



The 'Comparison type' dropdown menu is open, showing a list of options: 'case-insensitive' (selected), 'case-sensitive', 'pattern-rule', 'refer-case-sensitive', 'refer-case-insensitive', 'case-insensitive', and 'boolean'.

12. **Action**—Click the type of action you want to be performed in the combination box: **none**, **add**, **delete**, **manipulate**, **store**, **sip-manip**, and **find-replace-all**. These are the same actions you can select when configuring SIP header manipulation. The default is **none**.



The 'Action' dropdown menu is open, showing a list of options: 'sip-manip' (selected), 'replace', 'delete-element', 'reject', 'store', 'delete', 'delete-header', 'none', 'find-replace-all', 'manipulate', 'add', 'log', and 'sip-manip'.

13. **Match value**—Enter the value to match against the body part in the SIP message. This is where you can enter values to match using regular expression values. Your entries can contain Boolean operators.



The 'Match value' text input field contains the text 'match|'.

14. **Message type**—Click the SIP message type on which you want the MIME rules to be performed in the combination box. Valid values are **any**, **request**, and **reply**. The default value is **any**.

Message type

A dropdown menu with a blue border. The selected item is 'out-of-dialog'. The list of items includes 'out-of-dialog', 'any', 'request', and 'reply'.

15. **Format**—Enter the method for the Net-Net SBC to convert a specific parameter to a string representation of that value.

Format

A dropdown menu with a blue border. The selected item is 'ascii-string'. The list of items includes 'ascii-string', 'binary-ascii', and 'hex-ascii'.

16. **Content type**—Enter the content type for this MIME rule. This value refers to the specific body part in the SIP message body that is to be manipulated. For example, given a SIP Content-Type header with the value `multipart/mixed;boundary=unique-boundary-1`, you would enter a content-type value of **application/sdp** to manipulate specifically on the SDP portion of the SIP message.

To manipulate the SIP preamble or epilogue, enter the keyword **@preamble** or keyword **@epilogue**.

Content type

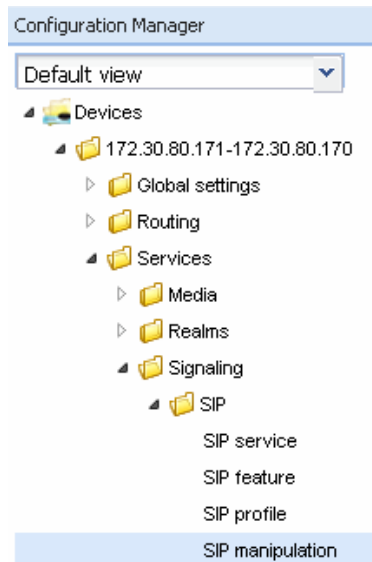
A simple rectangular text input field with a blue border.

17. Click **Apply**.

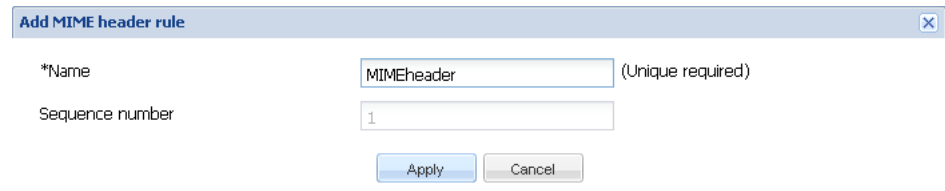
Creating MIME Headers

To create MIME headers:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select the SIP manipulation rule from the table and click **Edit**.
5. Scroll to the Rules element table.
6. Select the MIME rule from the table and click **Edit**.
7. Scroll to the MIME header rule table and click **Add**. The Add MIME header rule dialog box appears.
8. **Name**—Enter a name for this MIME header rule. This parameter is required and has no default.



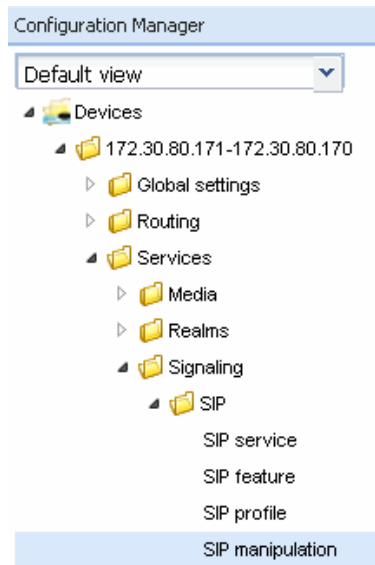
The dialog box titled "Add MIME header rule" contains two input fields. The first field, labeled "*Name", contains the text "MIMEheader" and has "(Unique required)" to its right. The second field, labeled "Sequence number", contains the value "1". At the bottom of the dialog are two buttons: "Apply" and "Cancel".

9. Click **Apply**. The MIME header rule is added to the MIME header rule table.
10. Repeat steps 8 through 10 to add additional MIME rules.
11. Click **Apply**.

Configuring MIME Headers

To configure MIME headers:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select the SIP manipulation rule from the table and click **Edit**.
5. Scroll to the Rules element table.
6. Select the MIME rule from the table and click **Edit**.
7. Select a MIME header rule in the MIME header rules table and click **Edit**.

8. **New value**—When the action parameter is set to **add** or to **manipulate**, click the new value that you want to substitute in the combination box.

New value

9. **Action**—Click the type of action you want to be performed in the combination box. The default is **none**.

Action

- replace
- delete-element
- reject
- store
- delete
- delete-header
- none
- find-replace-all
- manipulate
- add
- log
- sip-manip

10. **Comparison type**—Click the **comparison type** value in the combination box. This is the way that you want the body part of the SIP message to be compared. This choice dictates how the Net-Net SBC processes the match rules against the SIP header. The default is **case-sensitive**.

Comparison type

- case-sensitive
- pattern-rule
- refer-case-sensitive
- refer-case-insensitive
- case-insensitive
- boolean

11. **Match value**—Enter the value to match against the header in the body part of the SIP message. This is where you can enter values to match using regular expression values. Your entries can contain Boolean operators.

Match value

12. Click **Apply**.

About MIME ISUP Manipulation

MIME ISUP manipulation supports performing HMR operations on SIP ISUP binary bodies, and is configured in the **MIME ISUP rules** configuration. This configuration works the same way that the MIME rule configuration does and contains the same parameters for you to set, but it also includes additional parameters and a sub-configuration targeted specifically for ISUP application.

Net-Net SBC MIME ISUP Parameters

- **Message type**—Refers to specific ISUP message types (such as IAM and ACM). The Net-Net SBC uses with the **msg-type** parameter (which identifies the SIP message) in the matching process. You enter values in this parameter as a list of numbers rather than as an enumerated value because of the large number of ISUP message types, and the range is between 0 and 255.
- **ISUP specification**—Specifies how the Net-Net SBC is to parse the binary body; valid values are the enumerated type. The values for this parameter are the following SIP ISUP formats:
 - ANSI-2000—Corresponding to ANSI T1.113-2000
 - ITU-99—Corresponding to ITU Q.763

Because ISUP messages do not identify their format, you must designate which you want to use.

- **ISUP parameter rules** (sub-configuration)—Use to create, manipulate, and store different parameters in the body of the ISUP message. Two parameters for this rule are unique: **Parameter rule** and **Parameter format**.
 - **Parameter rule**—Using ISUP parameter, this setting identifies the ISUP parameters on which you want to perform manipulation. This parameter takes values between 0 and 255, and you must know the correct ISUP mapping value for your entry. The Net-Net SBC calculates the offset and location of this parameter in the body. The value returned from the body does not the type or length, only the parameter value. For example, a parameter-type value of 4 acts on the Called Party Number parameter value.

In accordance with the ISUP specifications, only certain message types are allowed to have optional parameters. If optional parameters are present, an offset field must exist for them; so its value is 0 even if there are no optional parameters in the SIP message. For example, if you define a SIP ISUP rule that applies to all message types and adds a parameter that is neither fixed nor variable: The Net-Net SBC adds it as an optional parameter even if message type should not support optional parameters.

If you define an ISUP parameter rule with an **add** action and an empty **new-value**, the Net-Net SBC uses the default for that parameter. If you define an ISUP parameter rule with a **replace** action and no parameters exist, the Net-Net SBC will not perform any action. This behavior is consistent with that of SIP header rules in that a value can only be replaced if it already exists. If there is a value and no new value is set, the Net-Net SBC set it as a zero-length parameter.

- **Parameter format**—This parameter converts the specific parameter to a string representation of that value. Valid values for **parameter format** are: **number-param**, **hex-ascii** (default), **binary-ascii**, **ascii-string**, and **bcd**.

Both **match value** and **new value** are encoded and decoded by the designated parameter-format type. In this regard, the **match value** decodes the parameters and the **new value** encodes the ASCII string into the respective binary format.

Note: If you enter a new-value setting larger than the size of the parameter, the Net-Net SBC will perform no operation and will generate a corresponding error log message.

The following table provides information about the values you can enter:

Parameter Format Setting	Description
hex-ascii	Default. Converts the entire binary body. Non-hexadecimal characters fail in matching against the body part if they are in the match-value setting and non-hexadecimal characters places in a new-value setting result in no operation being formed.
binary-ascii	Converts each hexadecimal value to its corresponding string, binary representation. For example, the Net-Net SBC would convert the ISUP parameters with a hexadecimal binary value of 8A to 10001010. Non-binary digit characters fail when matching against the body part if they are contained in the match-value setting and non-binary characters in the new-value setting results in no operation being performed.
ascii-string	Treats the binary parameter as true ASCII in raw format. The Net-Net SBC supports only the printable range of ASCII characters. If a value in the ISUP parameters cannot be decoded to ASCII, the Net-Net SBC returns an empty string. Non-printable or meta characters cannot be entered as new-value settings, so this presents no issues for encoding.

Parameter Format Setting	Description
bcd	In ISUP, BCD refers to the binary format of the number used as a half a byte nibble, with the byte's lower nibble containing the first digit and the higher containing the second digit. For example, the number 127 is encoded as the two binary bytes 0x2107 on the wire. Using this mode, the Net-Net SBC treats the binary ISUP content as BCD; it should decode it from 0x2107 to the string 1270, and from a string of 127 it should decode it as 0x2107. Because a byte has two nibbles, a nibble might have to be added. And when the Net-Net SBC performs decoding, it cannot know that a BCD byte represents one or two ASCII digits—so it assumes there are two. The number-param setting decodes the parameter as a common number parameter. The Net-Net SBC sees the odd/even bit as in the first bite as telling it how many nibbles to decode correctly, and it will set the odd-even when it decodes. Non-binary digit characters fail to match against the body part if they are contained in the match value, and non-binary characters in the new value results in no operation being performed.
number-param	As the decimal value of the specified number type, treats the parameters as a generic number parameter type. For example, a parameter-type 4 acts on the Called Party Number parameter. When the action type is replace or add, the Net-Net SBC automatically sets the parameter's odd-even bit based on the number being inserted in relation to the new-value setting. If the Numbering Plan Indicator bits are 0b001 (ISDN, E.164), then the Net-Net SBC sets the Nature of Address field to 0b0000100 (international). If this number type is added to a non-existent parameter field, then the Numbering Plan Indicator field is 0b0000011 (national number). If this number type is added to a non-existent parameter field, then the Numbering Plan Indicator field will be set to 0b001 (ISDN, E.164) and the Net-Net SBC will also follow the previous rules. Regardless of the action type you set, the string represented for match-value use for this type will be the numbers of the address fields after the BCD coding. There will be a leading plus sign (+) if the Number Plan is 0b001 and the Nature of Address is 0b0000100 ((international); otherwise, there will not be a plus sign (+). If it cannot convert the data field to a number parameter, the Net-Net SBC returns an empty string. And if the new-value is not in digit form or cannot fit in the specified parameter type field, the Net-Net SBC takes no action.

Adding an ISUP Body to a SIP Message

Unlike the MIME manipulation you can use by setting the SIP header rules accordingly, you can add MIME parts to SIP messages using the MIME rules configuration.

You can configure a SIP header manipulation to add an ISUP body to a SIP message, and the Net-Net SBC adds them after any SDP parts if they are present. You can add an ISUP body to a SIP message in two ways:

- Create a **MIME ISUP rule** with the **action** type set to **add**, and enter the entire body in string hexadecimal form in the **New value** parameter.
- Leave the **New value** parameter empty at the **MIME ISUP rule** level and create an add rule for an **ISUP parameter rule**.

In this case, the Net-Net SBC creates the corresponding ISUP message based on the **ISUP message type** value and supplies all of the parameters with their default values. Because the **ISUP message type** takes a list of values as a valid entry, for this case it only uses the first one. However, the Net-Net SBC ignores the **ISUP message type** value if you set the **new value** parameter. **ISUP parameter rule**, if configured, overwrites the default value or add a new parameter based on the defined parameter type.

It is also possible that you might supply a **new value** both at the **MIME ISUP rule** level and at the **ISUP parameter rule** level. If you do, the **new value** entry from the **MIME ISUP rule** is parsed into an ISUP object and the **ISUP parameter rule** operates on that object.

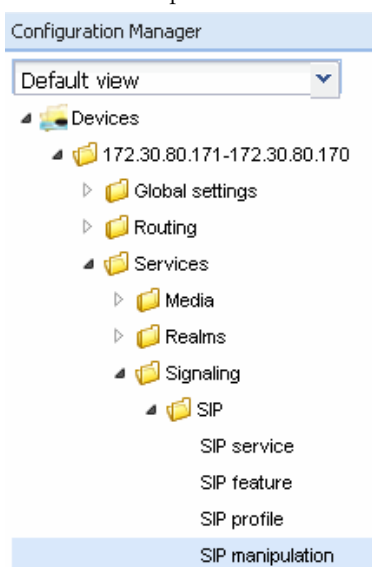
MIME ISUP Rules

These sections show you how to create and configure MIME ISUP rules.

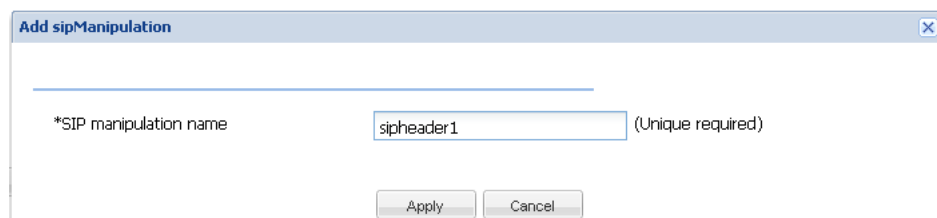
Creating MIME ISUP Rules

To create MIME ISUP rules:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.

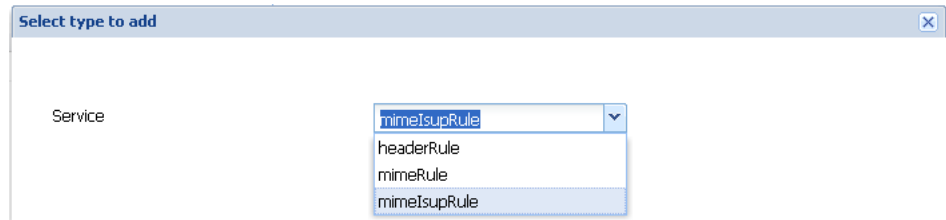


3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Click **Add**. The Add SIP manipulation rules dialog box appears.
5. **SIP manipulation name**—Type a SIP manipulation rule name as a unique identifier for this instance.

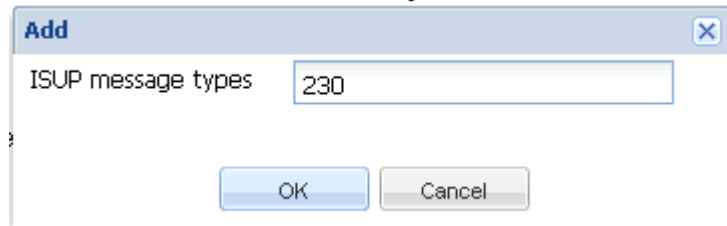


6. Click **Apply**.
7. Select the SIP manipulation rule from the table and click **Edit**.

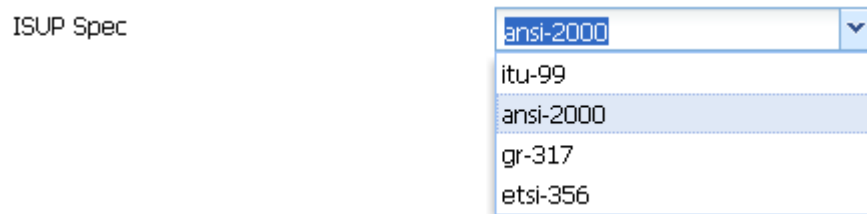
8. Scroll to the Rules element table.
9. Click **Add**. The Select type to add dialog box appears.
10. **Service**—Click **MIME ISUP Rules** in the combination box to designate this SIP manipulation rule a MIME ISUP rule.



11. Click **Apply**. The Add MIME ISUP rule dialog box appears.
12. **Name**—Enter a name for this MIME ISUP rule. This parameter is required and has no default.
13. Click **Apply**.
14. Select the MIME ISUP rule from the table and click **Edit**.
15. Scroll the ISUP message type table and click **Add**.
16. **ISUP message type**—Enter the specific ISUP message type (such as IAM and ACM) on which to operate. The Net-Net SBC uses the **ISUP message type** parameter (which identifies the SIP message) in the matching process. You enter values as a list of numbers rather than as an enumerated value because of the large number of **ISUP message types**. The valid values are between 0 and 255 and there is no default for this parameter.



17. Click **OK**. The ISUP message type is added to the ISUP message types list.
18. Repeat steps 16 through 19 to add additional **ISUP message types**.
19. **ISUP spec**—Specify how the Net-Net SBC is to parse the binary body; valid values are the enumerated type. The values for this parameter are these SIP ISUP formats:
 - **ANSI-2000** (default)—Corresponding to ANSI T1.113-2000
 - **ITU-99**—Corresponding to ITU Q.763



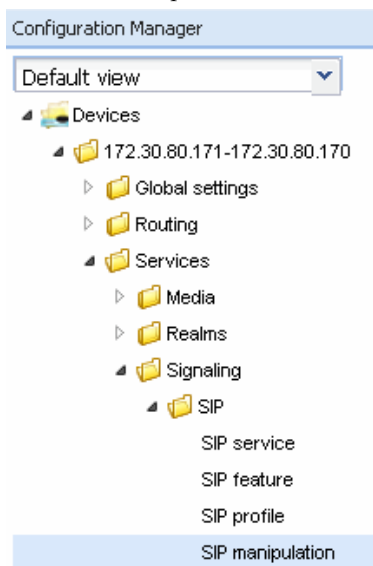
20. Click **Apply**. The MIME ISUP rule is added to the SIP manipulation table.
21. Repeat steps 9 through 21 to add additional MIME ISUP rules.

22. Click **Apply**.

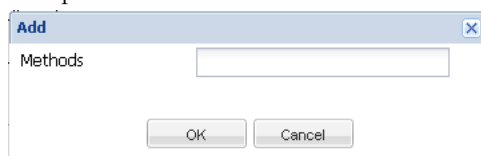
Configuring MIME ISUP Rules

To configure MIME ISUP rules:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select a SIP manipulation and click **Edit**.
5. In the Rules element table, select the MIME ISUP rule you want to configure and click **Edit**.
6. **Methods**—Click **Add**. The Add Methods dialog box appears.
7. Enter the SIP method to which the MIME rules applies. There is no default for this parameter.

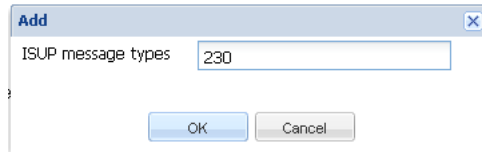


8. Click OK. The method is added to the Methods list.
9. Repeat steps 7 through 9 to add additional SIP methods.
10. **New value**—When the action parameter is set to **add** or to **manipulate**, click the new value that you want to substitute in the combination box.

New value

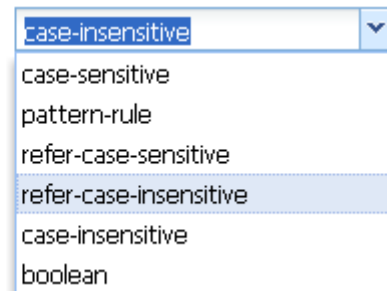
11. **ISUP messages types**—Click **Add**. The Add ISUP message type dialog box appears.
12. **ISUP message type** —Enter the number that specifies the specific ISUP message type (such as IAM and ACM) on which to operate. The Net-Net SBC uses the ISUP message type (which identifies the SIP message) in the matching

process. The valid values are between 1 and 255. There is no default for this parameter.



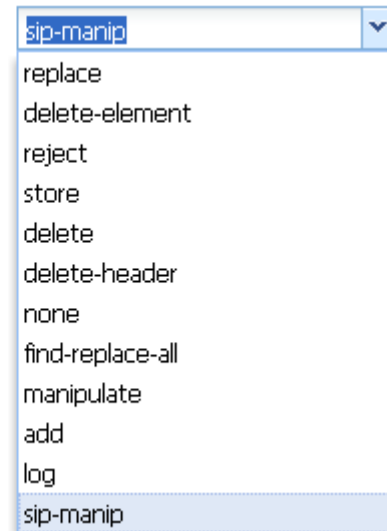
13. Click **OK**. The ISUP message type is added to the ISUP messages types list.
14. Repeat steps 12 through 14 to add multiple ISUP message types.
15. **Comparison type**—Click the comparison type value in the combination box. This is the way that you want the body part of the SIP message to be compared. This choice dictates how the Net-Net SBC processes the match rules against the SIP header. The default is **case-sensitive**.

Comparison type



16. **Action**—Click the type of action you want to be performed in the combination box. These are the same actions you can select when configuring SIP header manipulation. The default is **none**.

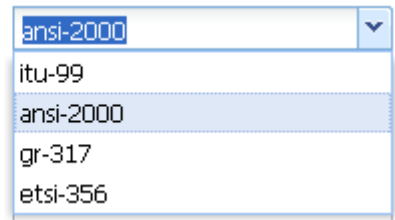
Action



17. **ISUP specification**—Click one of the following SIP ISUP formats in the combination box to indicate how the Net-Net SBC will parse the binary body:
 - **ANSI-2000** (default)—Corresponding to ANSI T1.113-2000

- ITU-99—Corresponding to ITU Q.763

ISUP Spec



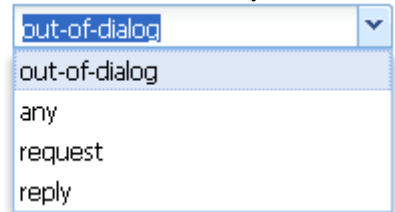
ansi-2000
itu-99
ansi-2000
gr-317
etsi-356

18. **Match value**—Enter the value to match against the body part in the SIP message. This is where you can enter values to match using regular expression values. Your entries can contain Boolean operators.

Match value

19. **Message type**—Click the SIP message type on which you want the MIME rules to be performed in the combination box. The default value is **any**.

Message type



out-of-dialog
out-of-dialog
any
request
reply

20. **Content type**—Enter the content type for this MIME rule. This value refers to the specific body part in the SIP message body that is to be manipulated. For example, given a SIP Content-Type header with the value `multipart/mixed;boundary=unique-boundary-1`, you would enter a content-type value of **application/sdp** to manipulate the SDP portion of the SIP message.

To manipulate the SIP preamble or epilogue, enter the keyword **@preamble** or keyword **@epilogue**.

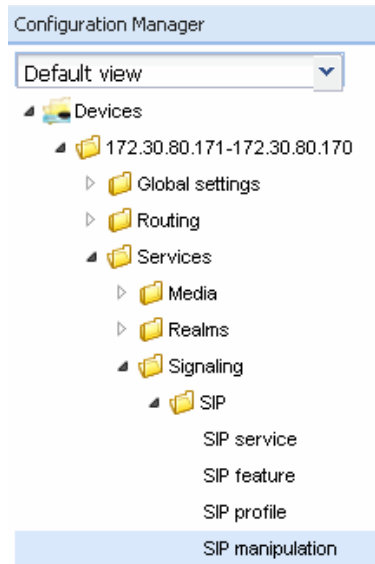
Content type

21. Click **Apply**.

Creating MIME ISUP Parameter Rules

To create ISUP parameters rules:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select a SIP manipulation and click **Edit**.
5. In the Rules element table, select the MIME ISUP rule you want to configure and click **Edit**.
6. Scroll to the Param rules and MIME header rules table and click **Add**.
7. **Service**—Select isupParamRule from the combination box and click **Apply**.
8. **Name**—Enter a name for this ISUP parameter rule and click **Apply**.
9. Select the ISUP Parameter rule from the table and click **Edit**.
10. **Parameter type**—Enter the ISUP parameters on which you want to perform manipulation. This parameter takes values between 0 and 255, and you must know the correct ISUP mapping value for your entry.

The Net-Net SBC calculates the offset and location of this parameter in the body. Note that the value returned from the body does not affect the type or length, only the parameter value. For example, a parameter type value of 4 acts on the **Called Party Number** parameter value.

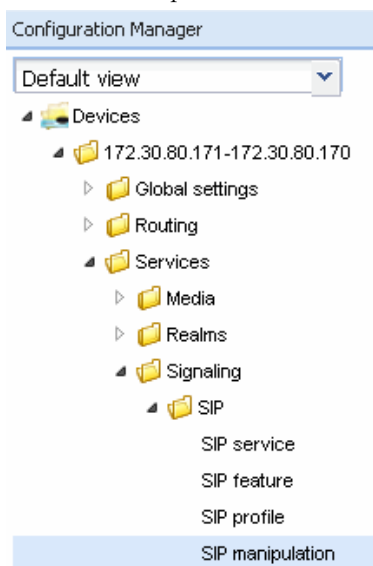
Parameter type

11. Click **Apply**. The ISUP parameter rule is added to the ISUP parameter rule table.
12. Repeat steps 7 through 12 to add multiple ISUP parameter rules.
13. Click **Apply**.

Configuring MIME ISUP Parameter Rules

1. Expand the Configuration Manager slider > Services > Signaling > SIP.

- Click SIP manipulation.



- Click the SIP manipulation name you want to configure and click **Edit**.
- Select a SIP manipulation and click **Edit**.
- In the Rules element table, select the MIME ISUP rule you want to configure and click **Edit**.
- Scroll to the Param rules and MIME header rules table and click **Edit**.
- New value**—When the action parameter is set to **add** or to **manipulate**, enter the new value that you want to substitute.

New value

- Parameter format**—Click the format in the combination box to convert specific parameter to a string representation of that value. The valid values are:

Both match value and new value are encoded and decoded by the designated parameter format type. In this regard, the match value decodes the parameters and the new value encodes the ASCII string into the respective binary format.

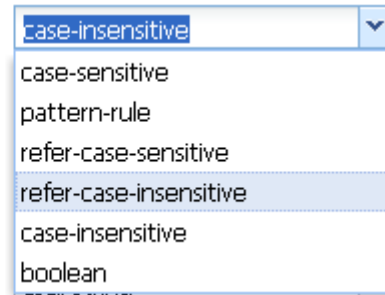
Format

binary-ascii
ascii-string
hex-ascii

- Comparison type**—Click the comparison type value in the combination box. This is the way that you want the header in the body part of the SIP message to

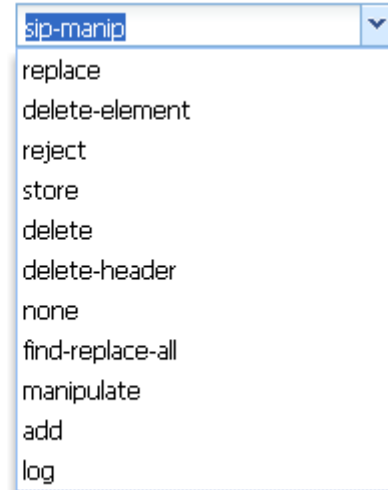
be compared. This choice dictates how the Net-Net SBC processes the match rules against the SIP header. The default is **case-sensitive**.

Comparison type

A dropdown menu with a blue border and a downward arrow on the right. The selected item is 'case-insensitive'. The list of items includes: case-insensitive, case-sensitive, pattern-rule, refer-case-sensitive, refer-case-insensitive, case-insensitive, and boolean.

10. **Action**—Click the type of action you want to be performed in the combination box. The default is **none**.

Action

A dropdown menu with a blue border and a downward arrow on the right. The selected item is 'sip-manip'. The list of items includes: sip-manip, replace, delete-element, reject, store, delete, delete-header, none, find-replace-all, manipulate, add, and log.

11. **Parameter type**—Enter the ISUP parameter on which you want to perform manipulation. You must know the correct ISUP mapping value for your entry. The valid values are between 0 and 255.

The Net-Net SBC calculates the offset and location of this parameter in the body. The value returned from the body does not include the type or length, only the parameter type value. For example, a parameter type of 4 acts on the called party number parameter value.

Parameter type

12. Click **Apply**.

SIP-ISUP Format Version Interworking

ISUP messages can be carried in SIP messages through either a standard body or through a multipart MIME encoded body. While ANSI and ITU are the two major groups, each contains many specific variants. To facilitate instances where two sides of a call use different versions, the Net-Net SBC supports interworking between the following SIP ISUP formats: ANSI, ITU, ETSI-356 (an ITU variant), and GR-317 (an ANSI variant). To do so, the Net-Net SBC can move, delete, and add parameters to various sections of the message.

The ISUP message version is determined by the content type of the SIP message or the MIME content type. When the base and version parameters do not match, the Net-Net SBC first uses the base parameter value to determine the format. If there is

no base, the Net-Net SBC then checks the version parameter. And if there is neither, the Net-Net SBC uses the **ISUP version** configured in the **SIP ISUP profile** configuration from the inbound realm, session agent, or SIP interface. Available values for that parameter are **ansi-2000**, **itu-99**, **gr-317**, or **etsi-356**. The Net-Net SBC considers unknown any value for the version that fails to match one of these or is missing.

Messages that contain an unknown ISUP format pass through the Net-Net SBC untouched. If there are operations to be performed on them, however, SIP ISUP HMR will take place. After the body has been converted, the Net-Net SBC updates both the base and version parameters of the content-type.

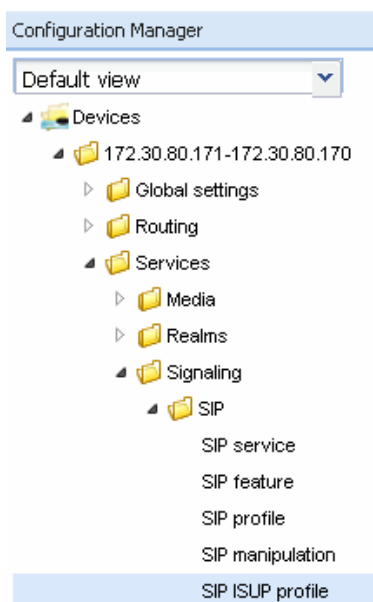
Custom formats are not supported.

Configuring SIP-ISUP Format Version Interworking

This section shows you how to set up a SIP ISUP format interworking. First, you configure a SIP ISUP profile, and then you apply it to a realm, session agent, or SIP interface.

To configure SIP-ISUP format version interworking:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



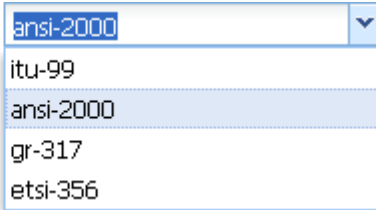
2. Click SIP ISUP profile. The SIP ISUP profile table appears in the content area.
3. Click **Add**. The Add SIP ISUP profile dialog box appears.
4. **Name**—Enter the name for the SIP ISUP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.

 A screenshot of the 'Add SIP ISUP profile' dialog box. It has a title bar with a close button. Inside, there is a label '*Name' followed by a text input field containing 'sipIsup1'. To the right of the input field is the text '(Unique required)'. At the bottom, there are two buttons: 'Apply' and 'Cancel'.

5. Click **Apply**.

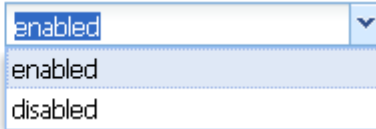
6. Click the SIP ISUP profile you want to configure and click **Edit**.
7. **ISUP version**—Specify the ISUP version you want used in this profile in order to support SIP-T. The default value is **ansi-2000**.

ISUP version



8. Click **Apply**.
9. **Convert outgoing messages to ISUP format**—Select enabled from the combination box to enable SIP ISUP format version interworking. The default is

Convert outgoing messages to ISUP format



disabled.

10. Click **Apply**.

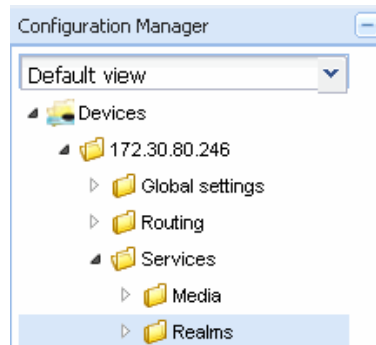
Applying SIP ISUP Profiles

When you want to enable this feature for a realm, session agent, or SIP interface, you configure the **SIP ISUP profile** parameter with the name of the profile you want applied.

The following sections show you how to apply SIP ISUP profiles to a realm, session agent, or SIP interface.

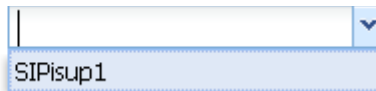
To apply a SIP profile and a SIP ISUP profile to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. **SIP ISUP profile**—Click the name of the SIP ISUP profile that defines the ISUP version to use for SIP Diversion SIP-ISUP interworking in the combination box. This is the value you entered in the **name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP Profile

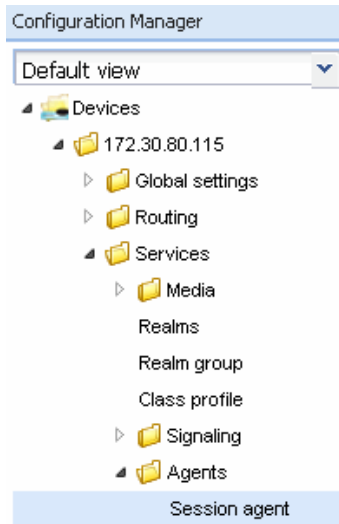


- Click **Apply**.

Session Agent

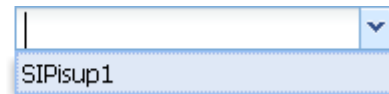
To apply a SIP profile and a SIP ISUP profile to a session agent:

- Expand the Configuration Manager slider > Services > Agents.



- Click Session agent. The Session agent table appears in the content area.
- SIP ISUP profile**—Click the name of the SIP ISUP profile that defines the ISUP version to use for SIP Diversion SIP-ISUP interworking. This is the value you entered in the **name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP Profile

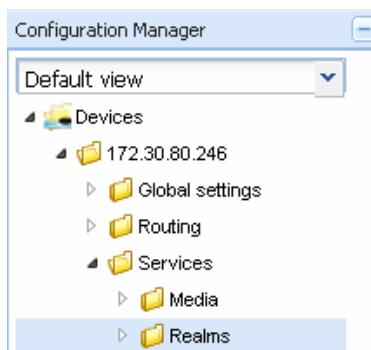


- Click **Apply**.

SIP Interface

To apply a SIP profile and a SIP ISUP profile to a SIP interface:

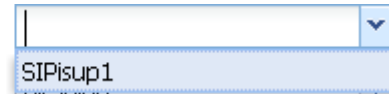
- Expand the Configuration Manager slider > Services.



- Click Realms. The Realm table appears in the content area.
- Select a SIP realm in the table and click **Edit**.
- Click the Services and Protocols tab. The Realm services table appears.

5. Select a SIP interface service in the table and click **Edit**.
6. **SIP ISUP profile**—Click the name of the SIP ISUP profile that defines the ISUP version to use for SIP Diversion SIP-ISUP interworking. This is the value you entered in the **name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP Profile



7. Click **Apply**.

SIP Proxy Subscriptions

When the Net-Net SBC operates in dialog mode (i.e., as a B2BUA), it creates and maintains dialog state for subscription dialogs created with SUBSCRIBE/NOTIFY messages and for INVITE-initiated dialogs. Since there can be a very large number of subscriptions per user in a Rich Communication Services (RCS) environment (especially for “presence” subscriptions), Net-Net SBC resources can quickly become depleted.

To alleviate this consumption of resources, you can configure your Net-Net SBC to operate in proxy mode for event packages that you define using the **Proxy sub events** parameter in the global SIP configuration. When you define event packages in this list and the operation mode for the SIP configuration is dialog or session, the Net-Net SBC processes all SUBSCRIBE and NOTIFY requests and responses for the designated event packages in transaction stateful mode.

Topology Hiding

So that it can perform topology hiding, the Net-Net SBC retains necessary routing information (such as the Contact or Record-Route header values) and it encodes certain data from the messages it receives in the messages it sends. To be more specific, the Net-Net SBC encodes the original URI hostport and the ingress realm name into a “gr” parameter it adds to the URI. The hostport information is replaced with the IP address and port of the SIP interface from which the message is sent (the egress interface). Without this information, subsequent in-dialog messages cannot be routed correctly because the Net-Net SBC does not retain dialog state (i.e., the route-set or remote-target).

For example, the URI `sip:td@192.168.24.121:5060` might be encoded as `sip:td@192.168.24.121:5060; gr=vjml9qtd175bhmhvkhgpp0jov81popvbp000040`.

The Net-Net SBC also performs URI encoding on the message body for Content-Type “application/pidf+xml”. This contains a Presence Information Data Format document (or PIDF) in PUBLISH and NOTIFY requests so subsequent SIP requests using the URIs in the PIDF document can be routed correctly.

The Net-Net SBC performs URI encoding in outgoing SIP messages after SIP=NAT is applied and before outbound HMR occurs. And the system decodes URIs in SIP messages after inbound HMR takes place and before SIP-NAT is applied.

In the event a URI is encoded several times (as is the case in spiral and hairpin calls), the encoded realm+hostport values are separated by a plus sign (+), as in the following:

`sip:td@192.168.24.121:5060; gr=vjml9qtd175bhmhvkhgpp+dhfhhb0jov81popvbp`

How It Works

When the Net-Net SBC receives any of the following requests, it matches the contents of the request's Event header with the list you configure in the proxy-sub-events parameter:

- PUBLISH
- SUBSCRIBE
- NOTIFY
- REFER

This is provided the operation-mode for the SIP configuration is set to either **session** or **dialog**. If it finds a match, the Net-Net SBC marks the request for processing in transaction-stateful mode rather than in B2BUA mode.

Note: Although PUBLISH is not a dialog-creating request, topology hiding needs to be applied to the PIDF so that subsequent NOTIFY requests containing portions of the published PIDF can be decoded properly.

When the Net-Net SBC forwards the request, it will have encoded all Contact and Record-Route header information using the ingress realm. The hostport value of the URIs then has egress SIP interface's IP address and port. The Via headers in the requested the Net-Net SBC received are not included in the outgoing request.

Then PUBLISH, SUBSCRIBE, NOTIFY, and REFER responses are compared to the request that was sent to determine if the response should receive transaction-stateful proxy treatment. The Net-Net SBC decodes any encoded Record-Route headers back to their original values for the outgoing response. Any Record-Route headers added downstream from the Net-Net SBC are encoded using the original request's egress realm (meaning the realm from which the response was received). In addition, the Contact header is encoded using the request's egress realm and ingress SIP interface and port.

Feature Interaction

When using this feature, the Net-Net SBC does not keep dialog or subscription state. Therefore the Per-user SUBSCRIBE Dialog Limit feature—configured in the **enforcement-profile** configuration—will not function properly when a subscription is handled in proxy mode.

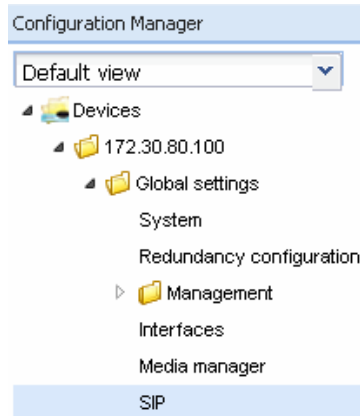
Configuring Proxy Sub events

This section shows you how to configure a list of SIP event packages to cause the Net-Net SBC to act in proxy mode.

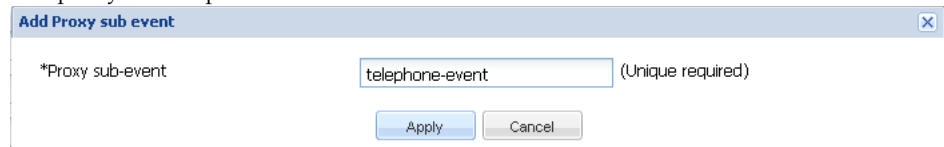
Note: The **Operation mode** parameter for the global SIP configuration must be set to either **dialog** or **session in order** for this feature to function as designed.

To configure SIP event packages to enable SIP proxy subscriptions:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to the Proxy sub event table.
5. Click **Add**. The Add proxy sub events dialog box appears.
6. **Name**— Enter the SIP event package name that you want to enable the SIP proxy subscriptions feature.



7. Click **Apply**. The proxy sub event is added to the Proxy sub event table.
8. Repeat steps 4 through 6 to add multiple proxy sub events.
9. Click **Apply**.

SIP Duplicate SDP Suppression

Using the **strip-dup-sdp** option in the SIP configuration, you can enable your Net-Net SBC to suppress a duplicate SDP answer in the reliable responses (1xx and 2xx) in an INVITE transaction.

During INVITE transactions in certain networks, SDP answers in reliable provisional responses (1xx) can cause interoperability issues. This issue occurs when the UAS includes the SDP answer in subsequent 1xx responses or in the final 2xx response, and the UAC views that inclusion as a protocol violation. The UAC does so based on the fact that the SDP answer was reliably delivered to it in a 1xx response, and so it

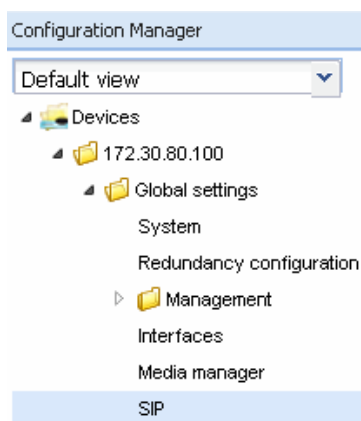
views additional SDP information as unnecessary in and ensuing reliable 1xx or 200 OK responses.

RFCs 3216 and 3262 do not specifically call out the UAS's including SDP information in this way as a protocol violation. Still, the Net-Net SBC allows you set enable the **strip-dup-sdp** option as a means of preventing the UAC from terminating sessions. With this option enabled, the Net-Net SBC removes the SDP answer in subsequent reliable provisional or final 200 OK responses if it is identical to the SDP answer previously received.

Configuring SIP Duplicate SDP Supression

To enable duplicate SDP suppression for SIP sessions:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to the Option section of the content area.
5. Click **Options**. The Options dialog box appears.
6. Click **strip-dup-sdp** in the Available options list to select it.
7. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.
8. Click **OK**.
9. Click **Apply**.

SIP LMSD Interworking

A flavor of SIP, the Legacy Mobile Station Domain (LMSD) supports existing Mobile Stations (MSs) in a network that supports IP bearers. LMSD uses the Alert-Info header in 180 Ringing responses to INVITEs to tell User Agents that specific tones (or ringback) are to be generated locally by the UAC instead of passing on media received from the UAS. When the 180 Ringing response includes SDP, most UAs that do not support LMSD ignore the Alert-Info and do not play ringback locally to the calling party; they expect in-band media from the far-end. As a result, the caller hears silence instead of ringing until the call is answered.

The Net-Net SBC's support for LMSD enables it to suppress SDP in the 180 Ringing response so the UAC will play local ringback. This feature also covers cases where the Alert-Info from an LMSD UAS specifies the busy/congestions tone.

You enable this feature for individual SIP interfaces that face LMSD user agents by setting the **lmsd-interworking** option. For example, if endpoints supporting LMSD care in the core realm, then you add the **lmsd-interworking** option to the core SIP interface.

About Alert-Info Processing

The content from an Alert-Info header can be a URI (in the form `<http://LMSD/tone?sig-id=tone>`) to indicate tone or in the form of `http://null` to indicate that media will be provided in-band from the far end. To account for variations in implementation, the Net-Net SBC looks only for **sig-id=** or for **/null** to determine the tone type. The following sig-id values yield the corresponding tones:

- **rt=ringtone**
- **bt=busy**
- **ct=congestion**

The Net-Net SBC treats all other sig-id values as ringtone.

When the Alert-Info contains neither **sig-id=** nor **/null**, the system behaves as though there were no Alert-Info. In other words, it ignores the Alert-Info.

About LMSD Processing

The Net-Net SBC uses LMSD on the egress side for an INVITE client transaction where the **lmsd-interworking** option is present on the egress SIP interface. Since LMSD endpoints are required to use reliable provisional responses, 100rel-interworking is used on the when the received INVITE does not contain a Supported or Require header with a 100rel option tag. In this case, the system adds the 100rel tag to the Supported header in the INVITE it sends toward the LMSD endpoint. And the system removes the Rack header and the 100rel tag from the response it sends to the UAC.

180 Ringing

Once the Net-Net SBC determines if the Alert-Info specifies ringtone, busy/congestion, or nothing, it proceeds in the following ways for each case:

- **Ringtone**—When the Alert-Info indicates or is treated as ringtone, the Net-Net SBC flags the client transaction as having an LMSD ringtone. It saves the SDP—if any is present—in the response in the client-side dialog and removes it from the response it sends to the UAC. The saved SDP might be used for the 2xx response to the INVITE. Also, the client-side dialog is marked for early SDP suppression to prevent the Net-Net SBC from forwarding an UPDATE request with a new SDP offer to the calling UA before a 2xx response arrives.
- **Busy/congestion**—When the Alert-Info indicates busy or congestion, the Net-Net SBC changes the 180 Ringing response to an error response. It uses 486 for busy and 503 for congestion. If the 180 Ringing is a reliable response, then the Net-Net SBC generates a PRACK to acknowledge it. The system will also send a CANCEL toward the UAS to cancel the INVITE transaction. The 486 or 503 error response is sent to the UAC after the removal of all message bodies and any Record-Route, Contact, Require: 100rel, or RSeq headers. The UAC should then generate the proper tone since it has not passed the UDP answer.

- Null—When there is no Alert-Info—or when the Net-Net SBC interprets there to be none—and the response contains SDP, LMSD is disabled for the remainder of the client transaction. And the Net-Net SBC will not check subsequent 180 responses for Alert-Info headers. The system will include an SDP answer in its response to the UAC, and the client-side dialog will no longer be marked for early SDP suppression.

Non-180 Responses

When it receives a on-180 provisional response with SDP, the Net-Net SBC disables LMSD for the remainder of the client transaction. No subsequent 180 responses will be checked for Alert-Info headers. Instead, the Net-Net SBC includes an SDP answer in the provisional response it sends to the UAC. In addition, the client-side dialog will no longer be marked for early SDP suppression.

UPDATE Request

If it sent an answer to the INVITE's SDP offer in a reliable provisional response, the UA can send a new SDP offer in an UPDATE request during an early dialog. In this case, "early" refers to a time before the INVITE 2xx response has been sent. If that SDP answer has been suppressed because of LMSD interworking, the Net-Net SBC must respond locally to the UPDATE in order to comply with SDP offer-answer rules. It cannot send the UPDATE to the calling UA.

Here, the Net-Net SBC checks to see if the dialog is marked for early SDP suppression. If it is, the Net-Net SBC generates a 200 OK response to the UPDATE and includes in it the SDP sent to the called UA in the original INVITE. The Net-Net SBC also saves the new SDP offer from the UPDATE in the client-side dialog, replacing any SDP saved previously. Then, if the SDP needs to be added to the INVITE 2xx response, it will be based on the latest SDP from the called UA.

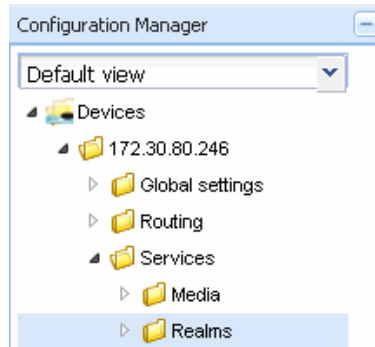
2xx INVITE Response

When a final success or 2xx response (usually a 200 OK) is received for the INVITE, it might not contain an SDP answer if the UAS had sent one previously in a reliable provisional response. If LMSD processing prevented an SDP answer from being sent to the UAC (in a reliable provisional response), an SDP answer must be added to the final 2xx response. If the SDP answer was saved in the client-side dialog, it will be added and the Net-Net SBC will update its flows accordingly. The resulting SDP answer will be part of the 2xx response destined for the UAC.

Configuring LMSD SIP Call Progress Tone Interworking

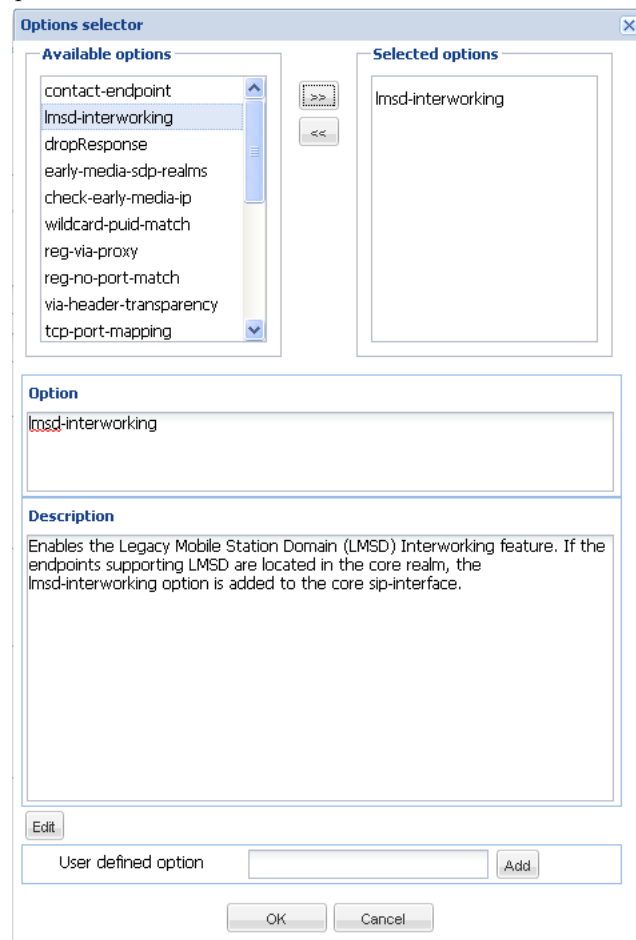
To enable SIP LMSD interworking for a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area. The Services and Protocols table appears.
5. Select the SIP interface and click **Edit**.
6. Click **Edit**.
7. Scroll to the Options in SIP service section and click **Options**. The Options dialog box appears.
8. Click **lmsd-interworking** in the Available options list to select it.

9. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** section contains a list box with the following items: contact-endpoint, lmsd-interworking (which is selected), dropResponse, early-media-sdp-realms, check-early-media-ip, wildcard-puid-match, reg-via-proxy, reg-no-port-match, via-header-transparency, and tcp-port-mapping. Between the two lists are two buttons: a right arrow (>>) and a left arrow (<<). The **Selected options** section contains a list box with the item lmsd-interworking. Below these sections, there is a section titled **Option** with a text field containing lmsd-interworking. Below that is a section titled **Description** with a text area containing the text: "Enables the Legacy Mobile Station Domain (LMSD) Interworking feature. If the endpoints supporting LMSD are located in the core realm, the lmsd-interworking option is added to the core sip-interface." Below the description is an **Edit** button. At the bottom, there is a section titled **User defined option** with a text field and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

10. Click **OK**.
11. Click **Apply**.

Comma-Separated SIP Header Values

Within a SIP message, multiple headers can exist within a single header if the values are separated by a comma. HMR does not have the capacity to handle these headers as unique header instances. Instead, it incorrectly treats the entire header as a single value.

Two new attributes under SIP manipulation have been added to address this issue: split headers and join headers. Each of these attributes are configured as a comma-separated list of header names.

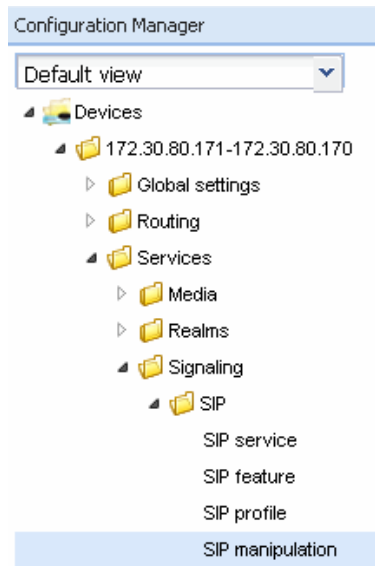
For each of the headers specified in the split headers list, all headers of that type with comma-separated values will be split into multiple headers with their values on separate lines.

For each of the headers specified in the join headers list, all headers of that type will be combined into a single header with their values separated by a comma.

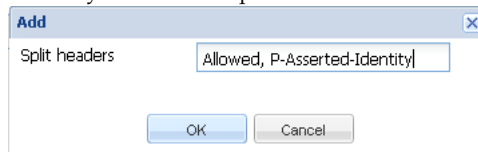
Configuring HMR Split Headers

To configure HMR split headers:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select the SIP manipulation name you want to configure and click **Edit**.
5. **Split headers**—Click **Add**. The Split headers dialog box appears.
6. Enter the name of the header that contains the multiple comma-separated values you want to split.

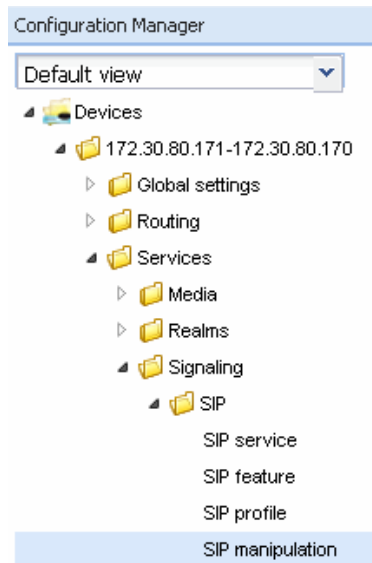


7. Click **OK**. The header name is added to the Split headers table.
8. Click **Apply**.

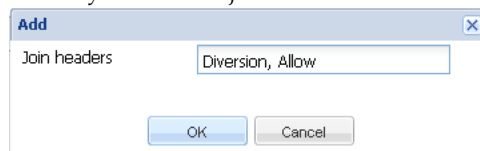
Configuring HMR Join Headers

To configure HMR join headers:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Select the SIP manipulation name you want to configure and click **Edit**.
5. **Join headers**—Click **Add**. The Join headers dialog box appears.
6. Enter the name of the header that contains the multiple comma-separated values you want to join.



7. Click **OK**. The header name is added to the Join headers table.
8. Click **Apply**.

Expedited Call Leg Release for Pre-empted Hairpin Calls

When hairpinned calls are ended because of signaling failures (such as a SIP mid-dialog signaling timeout, or an H.323 TCP keepalive failure) on one call leg, the Net-Net SBC deletes both legs' media flows simultaneously by default. In addition, when the first hairpinned call leg is torn down, the second call leg is gracefully released immediately by the Net-Net SBC creating and sending an appropriate signaling message (for example, BYE for a SIP call or ReleaseComplete for an H.323 call) to the endpoint.

You can override this behavior by configuring the **dont-terminate-assoc-legs** option in Media Control. When configured, the orphaned call leg in the hairpin scenario will be torn down after the **initial guard timer expires**. The disconnect times of the two call legs, as recorded by the accounting application, will be significantly different, due to the initial guard time for the second call leg.

Accounting Considerations

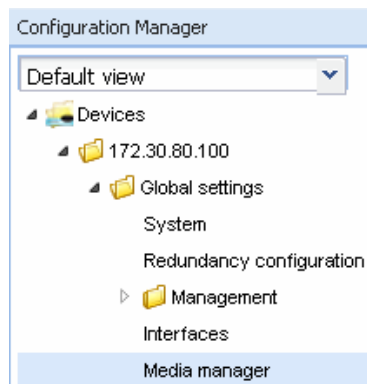
To indicate cases like this, where the second leg of the hairpinned call was preempted, Net-Net SBC includes the following combination of release/termination causes in the CDR:

```
VSA 49: Acct-Terminate-Cause = NAS_REQUEST
VSA 62: Acme-Disconnect-Cause = 8
```

Configuring Call Leg Release for Hairpin Calls

To configure expedited call leg release for hairpin calls:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to Options.
5. Click **Options**. The Options window appears.
6. Click **dont-terminate-assoc-legs** in the list of Available options to select it.
7. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
8. Click **OK**.
9. Click **Apply**.

Expedited Media Termination of Calls with Abnormal Signaling Termination

When a remote endpoint does not receive a BYE message from the endpoint hanging up, and/or the Net-Net SBC does not receive a 200 OK message in response to the BYE from the remote endpoint, the remote call leg's media session is not closed gracefully. In such a case, the media session is closed by a media timer expiring.

In a normal call flow, the Net-Net SBC closes the remote call leg when it receives the 200 OK back from the remote endpoint. In the aforementioned non-standard case, the Net-Net SBC never receives this 200OK. If such situations are expected, you can configure the Net-Net SBC to close the remote call leg's media session as it forwards the BYE message to the remote endpoint, and not wait any further. This is accomplished by configuring the **release-media-at-by** option in SIP configuration.

Accounting Considerations

Even if the endpoint that has been hung up on does not respond immediately, the CDR created for the call must have the stop time reflect when the BYE was received by the Net-Net SBC, not when the response was received or when the SBC timed out the abandoned call leg.

To indicate cases like this, where the call termination cannot be completed by standard means, the Net-Net SBC includes the following combination of release/termination causes in the CDR:

Acct-Terminate-Cause = Session-Timeout

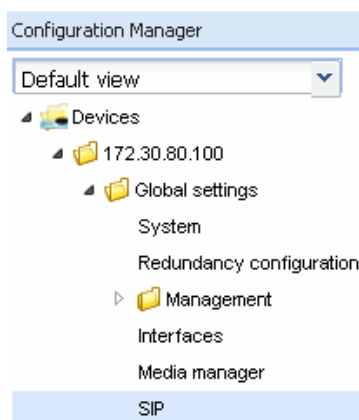
Acme-Disconnect-Cause = 47

Acme-SIP-Status = 908

Configuring Graceful Termination

To configure graceful termination when signaling ended abnormally:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The Options window appears.
5. Click **release-media-at-by** in the list of Available options to select it.

6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
7. Click OK.
8. Click Apply.

Wildcarded PUID in P-Associated-URI

In the IMS core, a PBX often registers its own base telephone number with the S-CSCF, and the S-CSCF obtains a wildcarded public user identity, represented by a regular expression that describes all extension numbers associated with the registering PBX.

The S-CSCF includes the wildcarded PUID in the 200OK response to the REGISTER message that the Net-Net SBC sent it, prompted by the initial registering endpoint. Upon receiving the 200OK, the Net-Net SBC uses the wildcarded puid to implicitly treat subsequent INVITEs from PBXs extensions that match the wildcard as registered endpoints.

For all calls directed at telephone numbers that are implicitly registered by the wildcarded puid regular expression, the Net-Net SBC terminates them the servicing PBX.

This behavior is configured by adding the **wildcard-puid-match** option on the access side SIP interface when requests originate there.

Wildcarded Public User Identity

A wildcarded PUI consists of a delimited regular expression located either in the userinfo portion of the SIP URI or in the telephone-subscriber portion of the Tel URI. The regular expression in the wildcarded PSI shall take the form of Extended Regular Expressions (ERE) as defined in chapter 9 in IEEE 1003.1-2004 Part 1. The delimiter is the exclamation mark character ("!").

Using the following PUI as an example:

```
sip:chatlist!.*!@example.com
```

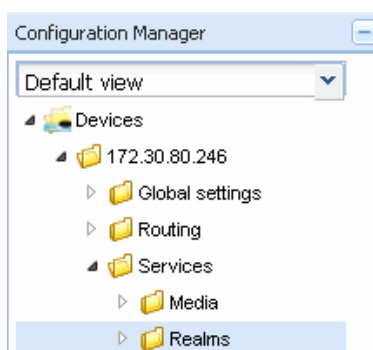
The following PUIs will match:

```
sip:chatlist1@example.com
sip:chatlist2@example.com
sip:chatlist42@example.com
sip:chatlistAbC@example.com
sip:chatlist!1@example.com
```


Configuring Wildcard PUID Registration

To configure wildcard PUID registration:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure in the Realm table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select the SIP interface you want to configure and click **Edit**.
6. Click the SIP interface tab.
7. Scroll to Options in SIP service.
8. Click **Options**.
9. Click **wildcard-puid-match** from the list of Available options to select it.
10. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
11. Click **OK**.
12. Click **Apply**.

Alarm Synchronization (or Trap Event Monitoring)

Two trap tables in the ap-smgmt.mib record trap information for any condition on the Net-Net SBC that triggers an alarm. You can poll these two tables from network management systems, OSS applications, and the Net-Net EMS to view the fault status on one of more Net-Net SBCs.

The two trap tables support alarm synchronization, and by polling them you can obtain information about the current fault condition on the Net-Net SBC. These tables are:

- apSysMgmtTrapTable-You can poll this table to obtain a summary of the Net-Net SBC's current fault conditions. The table records multiples of the same trap type that have occurred within a second of one another and have different information. Each table contains the following:
 - sysUpTime
 - Instance Number

- **apSysMgmtTrapInformationTable**—You can poll this table to obtain further details about the traps recorded in the **apSysMgmtTrapTable** table. The following information appears:
 - Data type
 - Data length
 - The data itself (in octets)

Trap Tables do not record information about alarm severity.

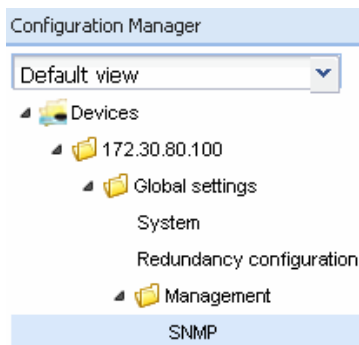
The **apSysMgmtTrapTable** can hold up to 1000 entries, and you can configure the number of days these entries stay in the table for a maximum of seven days. If you set this parameter to 0 days, the feature is disabled. And if you change the setting to 0 days from a greater value, then the Net-Net SBC purges the tables.

Note: The Net-Net SBC does not replicate alarm synchronization table data across HA nodes. Each Net-Net SBC in an HA node maintains its own tables.

Configuring Trap Event Monitoring

To configure trap event monitoring for alarm-triggering events:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click SNMP. The SNMP data window appears in the content area.
3. Scroll to Trap table.
4. **Trap event lifetime (days)**—Enter the number of days, from 1 to 7, you want to record trap information in the **apSysMgmtTrapTable** and the **apSysMgmtTrapInformationTable**. Leaving this parameter set to 0 (default) turns alarm synchronization off.

Trap event lifetime (days)

5. Click **Apply**.

Configuring CAC, Policing, and Marking for non-Audio, non-Video Media

You can enable the use of the AS modifier in the SDP and specify the percentage of headroom to be added.

Support for the AS Bandwidth Modifier

The following parameters provide support for AS bandwidth modifier:

- **SDP Bandwidth**—Enables or disables the use of the AS modifier in the SDP if the **Bandwidth requirements (Kbps)** and **SDP rate limit headroom (%)** EMS

labels are not set to valid values in a corresponding media profile. The default value is **disabled**.

- **SDP rate limit headroom (%)**—Specifies the percentage of headroom to be added while using the AS bandwidth parameter, while calculating the **Maximum bandwidth allowed per flow (bytes/sec)** (rate limit for the RTP flow). The default value is zero (0). The valid range is:
 - Minimum—0
 - Maximum—100

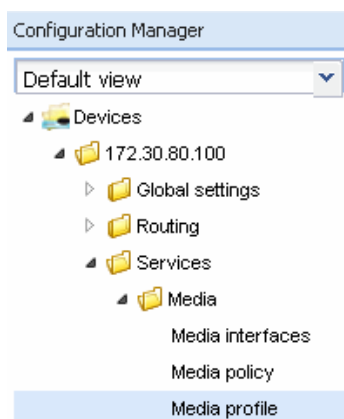
The following conditions apply to the use and application of these two parameters:

- If the amount of required bandwidth is not specified in the media profile (**Bandwidth requirements (Kbps)**) for the media type in the **m=** line of the SDP, then the value specified in the AS modifier is used. The Net-Net SBC only uses the AS value if you set the new **SDP Bandwidth** to **enabled**.
- If the average rate limit value for RTP flows is not specified in the media profile (**Maximum bandwidth allowed per flow (bytes/sec)**) for the media type in the **m=** line of the SDP, then the value specified in the AS modifier is used. The Net-Net SBC only uses the AS value if you set the new **SDP Bandwidth** to **enabled**. When calculating the average rate limit that it will use based on the AS modifier, the Net-Net SBC applies the percentage set in the **SDP rate limit headroom (%)** EMS label.
- The Net-Net SBC uses the value specified in the AS modifier (if **SDP Bandwidth** is enabled, and **Bandwidth requirements (Kbps)** is set to 0) along with the **Maximum bandwidth allowed** value set in the Realms window; this works the same way that the **Bandwidth requirements (Kbps)** parameter does.
- The Net-Net SBC uses the value specified in the AS modifier (if **SDP Bandwidth** is enabled, and **Bandwidth requirements (Kbps)** is set to 0) along with the **Bandwidth limit (kilobits per second)** value set in the Realms window; this works the same way that the **Bandwidth requirements (Kbps)** parameter does.

Enabling AS Modifier Support and Headroom

To enable AS modifier use and establish the percentage of headroom to use:

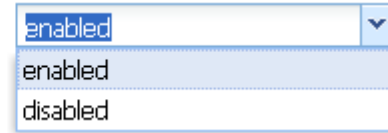
1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Select a media profile name in the table and click **Edit**. The Edit Media profile window appears.

4. **SDP Bandwidth**—Click the checkbox to enable this parameter to use the AS bandwidth modifier in the SDP. The default is disabled.

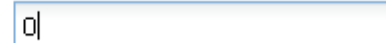
SDP bandwidth



5. **SDP rate limit headroom (%)**—Enter the percentage of headroom to be added while using the AS bandwidth parameter while calculating the **Maximum bandwidth allowed per flow (bytes/sec)** (rate limit for the RTP flow). The default is 0. The valid range is:

- Minimum—0
- Maximum—100

SDP rate limit headroom (%)



6. Click **Apply**.

Custom RADIUS CDR VSAs for SIP

This section describes the following additions to the Net-Net SBC's RADIUS accounting capabilities for customizing your CDRs:

- Generating CDRs with call detail information from a SIP message—The Net-Net SBC reserves a set of vendor-specific attributes (VSAs) and then populates them according to your header manipulation rules (HMR) configuration
- Generating CDRs with trunk group information—You can enable your Net-Net SBC to provide terminating trunk-group and trunk-context data even when the Net-Net SBC is not performing trunk-group routing.

Both support using the CSV file for RADIUS records, which you can either save locally or push to a defined FTP server.

About User-Defined VSAs for SIP Calls

The Net-Net SBC reserves VSAs 200-229 for you to define for use with SIP calls. These VSAs should never be used for other purposes, and their use should never conflict with the need to add new VSAs in the future. Because this leaves a significant number of VSAs unused, there is still ample space for any new VSAs that might be required.

Since RADIUS START records are created on session initiation, their content cannot be updated. However, the content for INTERIM and STOP records can be.

To configure user-defined VSAs for a SIP call, you use header manipulation rules (HMR). You can configure HMRs to let the Net-Net SBC report originating or terminating country codes in CDRs in whatever format they appear in the SIP username field. The HMRs you configure use the SIP header name P-Acme-VSA, adding it to the SIP header from any part of the SIP message. Then the Net-Net SBC searches for the P-Acme-VSA header, generates a VSA for it, and then includes that VSA in the CDR for the call.

You can include multiple custom VSAs per CDR by adding the corresponding number of rules; because in doing so you add in the header as many times as required.

HMR Adaptations

The following HMR element rule types support user-defined VSA for SIP calls:

- **uri-user-only**—Represents the URI username without the URI user parameters. You can perform these actions for the **uri-user-only** type: store, replaces, delete, and add. This means, for example, that you can add a username string to SIP or TEL URI without having any impact on other parameters.
- **uri-phone-number-only**—Applies when all rules are met. It refers to the user part of the SIP/TEL URI without the user parameters when the user qualifies for the BNF shown here:

```
uri-phone-number-only = [+]1*(phone-digit / dtmf-digit / pause-character)
phone-digit           = DIGIT / visual-separator
DIGIT                 = "0" / "1" / "2" / "3" / "4" / "5" / "6" / "7" / "8" / "9"
visual-separator      = "-" / "." / "(" / ")"
dtmf-digit            = "*" / "#" / "A" / "B" / "C" / "D"
pause-character       = "p" / "w"
```

Once the URI user part qualifies as a **uri-phone-number-only** based on this BNF, the Net-Net SBC ignores the visual separators when comparing it against a match value. Furthermore, the Net-Net SBC performs on or using the **uri-phone-number-only** after the excluding the visual separators.

But a new value being added as a **uri-phone-number-only** or replacing a **uri-phone-number-only** does not have to match the BNF noted above. That is, you can use the **uri-phone-number-only** type knowing that:

- The action only occurs if the URI username matches the BNF defined here
- Even so, you can also replace the **uri-phone-number-only** with one that does not match—using the same rule

HMR String Variable

HMR supports the use of a string variable that you can use to populate headers and elements. You set this value in the **manipulation string** parameter for a realm, SIP session agent, or SIP interface.

When a message arrives, the Net-Net SBC matches the string you provision to the closest session agent, realm, or SIP interface. The precedence for matching is in this order: session agent, realm, and then SIP interface. For example, the Net-Net SBC populates messages matching a session agent using the \$HMR_STRING variable, but it leaves the value empty for session agents that do not match.

You can use the string variable, for instance, for values specific to realms and session agents such as country code values when the regular expression pattern used to match a country code fails to do so.

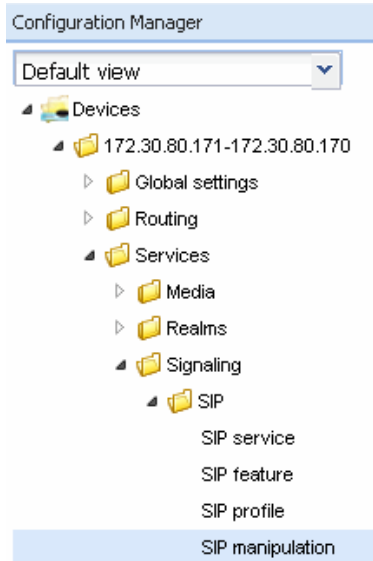
Configuring Custom CDR VSAs

This section shows you how to configure user-defined VSAs for SIP calls.

Creating a SIP Header Manipulation Ruleset

To configure SIP header manipulation ruleset:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. To add a SIP manipulation entity, click **Add**. The Add SIP header manipulation dialog box appears.
5. **SIP manipulation name**—Enter a name for the new SIP header manipulation ruleset. For example:

 A screenshot of the 'Add sipManipulation' dialog box. The title bar says 'Add sipManipulation'. Inside, there's a label '*SIP manipulation name' followed by a text input field containing 'sipheader1'. To the right of the input field is the text '(Unique required)'. At the bottom of the dialog, there are two buttons: 'Apply' and 'Cancel'.

6. Click **Apply** to save your new ruleset.
The new ruleset name appears at the top of the table.

Adding Header Rules

To add header rules

1. Click the new ruleset name in the table and click **Edit**.
2. Click **Add**. The Add Header rule dialog box appears.

- In the Service field, select headerRule from the combination box.
The Add Header, MIME and MIME ISUP rules dialog box appears.

- Header rule name**—Enter the name of the header to which this rule applies. (The name must match the SIP header name to which this rule applies.)
- Header name**—Enter the name of this rule.
- Click **Apply**. The header rule appears in the table.
- Select the header rule name and click **Edit**.
- Action**—Select the action you want applied to the header from the combination box.
- Comparison type**—Select the comparison type that the match value uses.
- Match value**—Enter the value to be matched, if any.
- Message type**—Select the message type to which this header rule applies from the combination box.

- Click **Apply**.

Creating Element Rules

To create element rules:

- In the Element rules area, click **Add**. The Add Element rule dialog box appears.
- Element rule name**—Enter the name of the element to which this rule applies.
Note: The name you enter depends on the element type you enter in step 5. For uri-param, uri-user-param, and header-param it is the parameter name to be added, replaced, or deleted. For all other types, it serves to identify the element rule and any name can be used.
- Click **OK**. The element rule appears in the table.
- Click the element rule and click **Edit**.

5. **Type**—Click the type of element on which to perform the action in the combination box. The options are:

Option	Description
header-value	Entire value of the header.
header-param	Parameter portion of the header.
uri-user	User portion of the SIP URI.
uri-host	Host portion of the SIP URI.
uri-port	Port number portion of the SIP URI.
uri-param	Parameter included in the SIP URI.
uri-header	Header included in a request constructed from the URI.
uri-user-param	User parameter of the SIP URI.

6. **Action**—Click the action you want applied to the element if there is a match value in the combination box. The options are:

Option	Description
add	Add a new element, if it does not already exist.
replace	Replace the elements.
delete-element	Delete the specified element, if it exists. Based on the match value, if entered in step 6.
delete-header	Delete the specified header, if it exists.

7. **Comparison type**—Click the type of value that needs to be matched for the action to be performed in the combination box. The options are:

Option	Description
case-sensitive	Perform a case-sensitive comparison of the stored value against the header.
case-insensitive	Perform a case-insensitive comparison of the stored value against the header.
pattern-rule	Prior to version S-C6.2, evaluate a referenced value by name to determine if its regular expression matches TRUE or FALSE. For release S-C6.2, the Net-Net SBC performs a regular expression comparison. If you are upgrading, this means that existing configurations where pattern-rule was set will be changed to boolean if the Net-Net SBC can parse the match value in the rule.
boolean	Evaluate a referenced value by name to determine if its regular expression or Boolean expression matches TRUE or FALSE. Also set the boolean setting on a specific stored match to determine whether or not that match has a stored value.

Option	Description
refer-case-sensitive	For release S-C6.2, perform a case-sensitive comparison of the stored value (or pattern rule subgroup) against: a header parameter value, URI parameter value, or header value.
refer-case-insensitive	For release S-C6.2, perform a case-insensitive comparison of the stored value (or pattern rule subgroup) against: a header parameter value, URI parameter value, or header value.

8. **Match value**—Enter the value you want to match against the element value for an action to be performed.
9. **Match value type**—Click the type of value in the combination box that needs to be matched to the match-field entry for the action to be performed. The default value is **ANY**. The values are:

Option	Description
IP	Element value in the SIP message must be a valid IP address to be compared to the match-value field entry. If the match-value field is empty, any valid IP address is considered a match. If the element value is not a valid IP address, it is not considered a match.
FQDN	Element value in the SIP message must be a valid FQDN to be compared to the match-value field entry. If the match-value field is empty, any valid FQDN is considered a match. If the element value is not a valid FQDN, it is not considered a match.
ANY	Element value in the SIP message is compared with the match-value field entry. If the match-value field is empty, all values are considered a match.

10. **New value**—Enter the value for a new element or to replace a value for an existing element. You can enter an expression that includes a combination of absolute values, pre-defined parameters, and operators.
 - Absolute values, with which you can use double quotes for clarity. You must escape all double quotes and back slashes that are part of an absolute value, and enclose the absolute value in double quotes.

For example:

```
"sip:"+$TRUNK_GROUP+"."+$TRUNK_GROUP_CONTEXT
```

 - Pre-defined parameters always start with a \$. Valid pre-defined parameters are:

Parameter	Description
\$ORIGINAL	Original value of the element is used.
\$LOCAL_IP	IP address of the SIP interface on which the message was received for inbound manipulation; or sent on for outbound manipulation.
\$REMOTE_IP	IP address the message was received from for inbound manipulation; or being sent to for outbound manipulation.
\$REMOTE_VIA_HOST	Host from the top Via header of the message is used.

Parameter	Description
\$TRUNK_GROUP	Trunk group is used.
\$TRUNK_GROUP_CONTEXT	Trunk group context is used.

- Operators are:

Operator	Description
+	Append the value to the end. For example: "acme"+"packet" generates "acmepacket"
+^	Prepends the value. For example: "acme"+"^"packet" generates "packetacme"
-	Subtract at the end. For example: "112311"."11" generates "1123"
.^	Subtract at the beginning. For example: "112311"."^"11" generates "2311"

- Click **Apply** to save your edits.

[sipManipulation->headerRule->elementRule](#)

*Element rule name	acme
Parameter name	\$EMPTY
Type	uri-param ▼
Action	add ▼
Comparison type	case-sensitive ▼
Match value	
Match value type	any ▼
New value	

The new element rule appears in the Element rules table.

- Click **Apply** to return to the Edit SIP manipulation ruleset window.

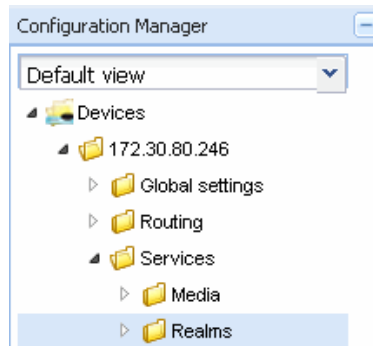
Configuring Manipulation Strings

To use the manipulation string variables, you set the **Manipulation string** value in the realm, SIP session agent, or SIP interface where you want the feature applied.

Realm

To configure manipulation strings for realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. Scroll to Manipulation string.
5. **Manipulation string**—Enter a value that references the \$HMR_STRING variable that will be used to populate SIP headers and elements using HMR. There is no default value for this parameter.

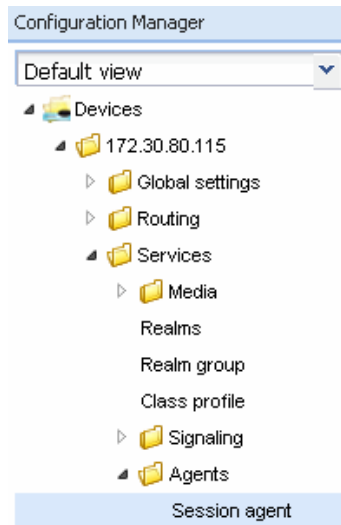
Manipulation string

6. Click **Apply**.

Session Agent

To configure manipulation strings for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.

3. Click a SIP session agent in the table and click **Edit**. The Edit session agent window appears.
4. Click the SIP session agent tab.
5. Scroll to Manipulation string.
6. **Manipulation string**—Enter a value that references the \$HMR_STRING variable that will be used to populate SIP headers and elements using HMR. There is no default value for this parameter.

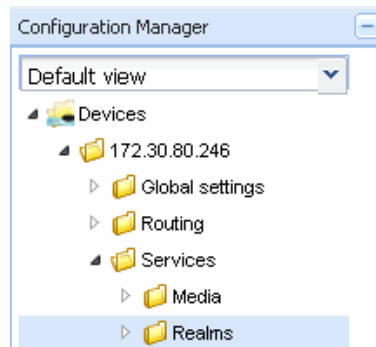
Manipulation string

7. Click **Apply**.

SIP Interface

To configure manipulation strings in a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP interface realm in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select a SIP interface from the table and click **Edit**.
6. Scroll to Manipulation string.
7. **Manipulation string**—Enter a value that references the \$HMR_STRING variable that will be used to populate SIP headers and elements using HMR. There is no default value for this parameter.

Manipulation string

8. Click **Apply**.

Trunk-Group VSA Generation

You can force the Net-Net SBC to generate VSAs related to trunk groups even when you are not using the trunk group feature. With the **force-report-trunk-info** parameter turned on in the routing profile tab, the Net-Net SBC reports:

- terminating trunk group and trunk context information even though it has not performed trunk group routing. The appropriate VSAs report the terminating trunk group (VSA 65) and trunk context (VSA 67) with the information of the matching ingress session agent and realm of the originator.
- terminating trunk group (VSA 66) and trunk context (VSA 68) as the received trunk group and context from the call's SIP REQUEST message. If the SIP message has none, then the Net-Net SBC uses the information from the

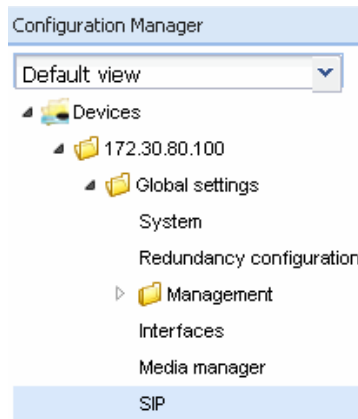
matching egress session agent (or egress realm, when available) and next-hop realm.

Note that information is reported after HMR processing—meaning that header manipulation has been performed on the message information reported.

Enabling Forced Trunk Group VSA Generation

To enable forced trunk group VSA generation:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **Force report trunk information**—Click the checkbox to enable **Force report trunk info**. By default, this parameter is **disabled**.
5. Click **Apply**.

SIP HMR MIME Support

Using the SIP HMR feature set, you can manipulate MIME types in SIP message bodies. While you can manipulate the body of SIP messages or a specific content type using other iterations of SIP HMR, this version gives you the power to change the MIME attachment of a specific type within the body by using regular expressions. To achieve this, you use the **find-replace-all** action type, which enables the search for a particular string and the replacement of all matches for that type. Although you use **find-replace-all** to manipulate MIME attachments, it can also be used to achieve other goals in SIP HMR.

Note that using **find-replace-all** might consume more system resources than other HMR types. Therefore this powerful action type should only be used when another type cannot perform the type of manipulation you require.

xx action type has recursive nature; will look until it can't find any more instances of what it's looking for .

Availability

Available in nnC511p10

How It Works: Find and Replace All

To manipulate a particular portion of the MIME attachment—as is the case for removing a certain attribute within the content type of **application/sdp**—the Net-Net SBC would need to search the content multiple times because:

- SDP can have more than one media line, and
- The SIP message body can contain more than one **application/sdp**.

The **find-replace-all** action type works for SIP header rules and for element rules. You can use it for all manipulation types from the entire header value, to the URI specific parameters, to MIME attachment.

For this action type, it does not matter what you configure the comparison type—which is atypical for actions types, as the comparison type is vital to the others. **Find-replace-all**, however, binds the comparison type to the pattern rule. Thus, the Net-Net SBC treats the match value as a regular expression, and it ignores any configured comparison type value in favor of the pattern rule. This type of action is both a comparison and action: For each regular expression match within the supplied string, the Net-Net SBC substitutes the new value for that match. Yet if you want to replace a certain portion of the regular expression and not the entire matched expression, you need to use a subgroup of expressions and the right syntax to indicate the sub-group replacement index.

You can indicate the sub-group replacement syntax by adding the string `[[ :n : ]]` to the end of the regular expression—where *n* is a number between 0 and 9. For example, given the following settings:

- Action=find-replace-all
- Match value=sip:(user)@host[[:1 :]]
- New value=bob

you create a new rule to replace only the user portion of the URI that searches for the regular expression and replaces all instances of the user subgroup with the value **bob**.

Taking advantage of the **find-replace-all**'s recursive nature, you can replace all the 0 digits in a telephone number with 1:

- Action=find-replace-all
- Match value=0
- New value=1

So for the user portion of a URI—or for any other string—with a value **1-781-308-4400** would be replaced as **1-781-318-4411**.

If you leave the **New value** parameter blank for **find-replace-all**, the Net-Net SBC replaces the matched sub-group with an empty string—an equivalent of deleting the sub-group match. You can also replace empty sub-groups, which is like inserting a value within the second sub-group match. For example, `user()@host.com[:1:]` with a configured **New value** `_bob` yields `user_bob@host.com`.

When you use **find-replace-all**, you cannot use the following **Type** values: **uri-param-name**, **uri-header-name**, and **header-param-name**. These values are unusable because the Net-Net SBC only uses case-sensitive matches for the match-value to find the parameter name within the URI. Since it can only be found by exact match, the Net-Net SBC does not support finding and replacing that parameter.

Escaped Characters

SIP HMR's support for escaped characters allows for searches for values you would be unable to enter yourself. Because they are necessary to MIME manipulation, support for escaped characters now includes:

- \f
- \n
- \r
- \t
- \v

New Reserved Word

To allow you to search for carriage returns and new lines, the SIP HMR MIME feature also adds support for the reserved word **\$CRLF**. Because you can search for these value and replace them, you also must be able to add them back in when necessary. Configuring **\$CRLF** in the **New value** parameter always resolves to `/r/n`, which you normally cannot otherwise enter through the ACLI.

About the MIME Value Type

Introduced to modify the MIME attachment, SIP HMR supports a **mime** value for the **Type** parameter in the element rules configuration. Like the **status-code** and **reason-phrase** values, you can only use the **mime** type value against a specific header—which in this case, is Content (abbreviated as **c**).

When you set the element rule type to **mime**, you must also configure the **Parameter name** with a value. This step is a requirement because it sets the content-type the Net-Net SBC manipulates in a specific part of the MIME attachment. You cannot leave this parameter blank; the Net-Net SBC does not let you save the configuration if you do. When you use the **store** action on a multi-part MIME attachment that has different attachment types, the Net-Net SBC stores the final instance of the content-type because it does not support storing multiple instances of element rule stored values.

In the event you do not know the specific content-type where the Net-Net SBC will find the **Match value**, you can wildcard the **Parameter name** by setting with the

asterisk (*) as a value. You cannot, however, set partial content-types (i.e., `application/*`). So configured, the Net-Net SBC loops through the MIME attachment's content types.

You can set the additional **Action** types listed in this table with the described result:

Action Type	Description
<code>delete-element</code>	Removes the matched mime-type from the body. If this is the last mime-type within in message body, the Net-Net SBC removes the Content-Type header.
<code>delete-header</code>	Removes all body content and removes the Content-Type header.
<code>replace</code>	Performs a complete replacement of the matched mime-type with the new-value you configure.
<code>find-replace-all</code>	Searches the specifies mime-type's contents and replaces all matching regular expressions with the new-value you configure
<code>store</code>	Stores the final instance of the content-type (if there are multi-part MIME attachments of various attachment types)
<code>add</code>	Not supported

MIME manipulation does not support manipulating headers in the individual MIME attachments. For example, the Net-Net SBC cannot modify the Content-Type given a portion of a message body like this one:

```
--boundary-1
Content-Type: application/sdp

v=0
o=use1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 10000 RTP/AVP 8
a=rtmap:8 PCMA/8000/1
a=sendrecv
a=ptime:20
a=maxptime:200
```

Back Reference Syntax

You can use back reference syntax in the **New value** parameter for header and element rules configurations. Denoted by the use of `$1`, `$2`, `$3`, etc. (where the number refers to the regular expression's stored value), you can reference the header and header rule's stored value without having to use the header rule's name. It instead refers to the stored value of "this" rule.

For example, when these settings are in place:

- Header rule=`changeHeader`
- Action=`manipulate`
- Match value=`(.+)([^\;])`

you can set the **New value** as `sip:$2` instead of `sip:$changeHeader.$2`.

You can use the back reference syntax for:

- Header rule **Actions** **manipulate** and **find-replace-all**
- Element rule **Actions** **replace** and **find-replace-all**

Using back reference syntax simplifies your configuration steps because you do not need to create a store rule and then manipulate rule; the manipulate rule itself performs the store action if the **Comparison type** is set to **Pattern rule**.

Notes on the Regular Expression Library

In the regular expression library, the dot (.) character no longer matches new lines or carriage returns. Conversely, the not-dot does match new lines and carriage returns. This change provides a safety mechanism preventing egregious backtracking of the entire SIP message body when there are no matches. Thus, the Net-Net SBC reduces backtracking to a single line within the body. In addition, there is now support for:

Syntax	Description
\s	Whitespace
\S	Non-whitespace
\d	Digits
\D	Non-digits
\R	Any \r, \n, \r\n
\w	Word
\W	Non-word
\A	Beginning of buffer
\Z	End of buffer
\	Any character including newline, in the event that the dot (.) is not

In addition, there is:

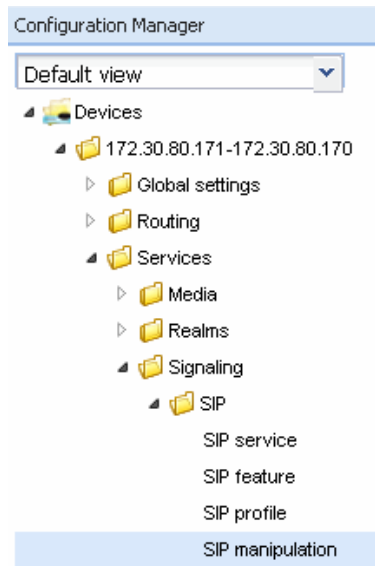
- Escaped character shortcuts (\w\W\S\s\d\D\R) operating inside brackets [...]

The **find-replace-all** action has been added to the header rules. Element rules support the **find-replace-all** action and the **mime** type.

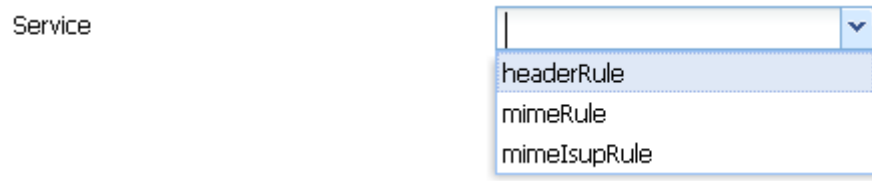
Configuring a SIP Header Rule

To set the header rule with the find-replace-all action:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Scroll to Rule elements and click **Add**. The Select type to add dialog box appears.
5. **Service**—Click **headerRule** in the drop-down list.

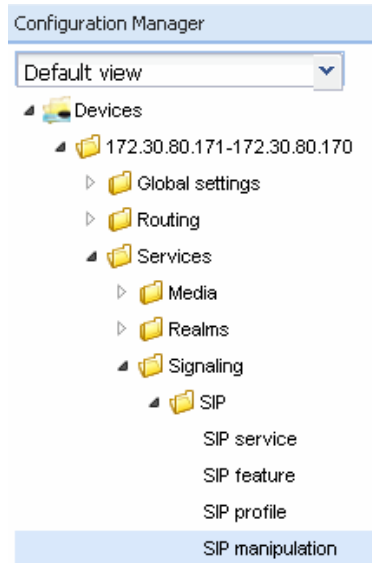


6. Click **Apply**. The Add Rule elements dialog box appears.
7. **Header rule name**—Enter the name for this header rule.
8. Click **Apply**.
9. A Success dialog box appears.
10. Click **OK**. The new rule element is added to the Rule element table.
11. Select the Header rule from the table and click **Edit**.
12. **Action**—Set the action parameter to **find-replace-all** if you want to enable SIP HMR MIME manipulation.
13. Click **Apply**.

Configuring a SIP Header Element Rule

To set the element rule with the find-replace-all action and MIME type:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Scroll to the Rule elements table.
5. Select the Header rule from the table and click **Edit**.
6. Scroll to the Element rule table and click **Add**.
7. **Element rule name**—Enter the name of the element to which this rule applies.
8. **Type**—Set the type parameter to **mime** if you want to enable SIP HMR MIME manipulation.
9. Click **Apply** in the Element rules window and click Back.
10. Click **Apply** in the Header rules window.

Configuration Examples

This section provides configuration examples based on the following sample SIP INVITE:

```
INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
```

```

c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 18
a=rtpmap:8 G729/8000/1
a=fmtp:18 annexb=no
a=sendrecv
a=ptime:20
a=maxptime:200

--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263a/90000
a=ptime:30

--boundary--

```

Example 1: Remove a Line from SDP

In this example, the SIP manipulation is configured to remove all p-time attributes from the SDP.

```

sip-manipulation
  name                                removePtimeFromBody
  description                        removes ptime attribute from all bodies
  header-rule
    name                                CTypeManp
    header-name                        Content-Type
    action                              manipulate
    comparison-type                    case-sensitive
    match-value
    msg-type                            request
    new-value
    methods                            INVITE
  element-rule
    name                                remPtime
    parameter-name                      application/sdp
    type                                mime
    action                              find-replace-all
    match-val-type                      any
    comparison-type                    case-sensitive
    match-value                        a=ptime:[0-9]{1,2}(\n|\r\n)
    new-value

```

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```

INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70

```

```
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 18
a=rtpmap:18 G729/8000/1
a=fmtp:18 annexb=no
a=sendrecv
a=maxptime:200

--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263a/90000

--boundary-
```

Example 2: Back Reference Syntax

In this sample of back-reference syntax use, the goal is to change the To user. The SIP manipulation would be configured like the following:

sip-manipulation	
name	changeToUser
description	change user in the To header
header-rule	
name	ChangeHeader
header-name	To
action	manipulate
comparison-type	case-sensitive
match-value	
msg-type	request
new-value	
methods	INVITE
element-rule	
name	replaceValue
parameter-name	
type	header-value
action	replace
match-val-type	any
comparison-type	pattern-rule
match-value	(.+) (service) (.+)
new-value	\$1+Bob+\$3

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```

INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:Bob@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
...
...
...

```

Example 3: Change and Remove Lines from SDP

In this sample of changing and removing lines from the SDP, the goal is to convert the G.729 codec to G.729a. The SIP manipulation would be configured like the following:

```

sip-manipulation
  name
  description
  proprietary codec name
  header-rule
    name
    header-name
    action
    comparison-type
    match-value
    msg-type
    new-value
    methods
    element-rule
      name
      parameter-name
      type
      action
      match-val-type
      comparison-type
      match-value
      new-value
  std2prop-codec-name
  rule to translate standard to
  CTypeManp
  Content-Type
  manipulate
  case-sensitive
  any
  g729-annexb-no-std2prop
  application/sdp
  mime
  find-replace-all
  any
  case-sensitive
  a=rtpmap:[0-9]{1,3}(G729/8000\1\r\na=fmtp:[0-9]{1,3} annexb=no)[[:1:]]
  G729a/8000/1

```

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```

INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466

```

```
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 8
a=rtpmap:18 G729a/8000/1
a=sendrecv
a=maxptime:200

--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263a/90000

--boundary-
```

Example 4: Change and Add New Lines to the SDP

In this sample of changing and adding lines from the SDP, the goal is to convert non-standard codec H.263a to H.263. The SIP manipulation would be configured like the following:

sip-manipulation		
name	description	prop2std-codec-name
to standard codec name		rule to translate proprietary
header-rule		
name		CodecManp
header-name		Content-Type
action		manipulate
comparison-type		case-sensitive
match-value		
msg-type		any
new-value		
methods		
element-rule		
name		H263a-prop2std
parameter-name		application/sdp
type		mime
action		find-replace-all
match-val-type		any
comparison-type		case-sensitive
match-value		a=rtpmap:([0-
9]{1,3}) H263a/. *\\r\\n		
new-value		a=rtpmap:+\$1+"
H263/90000"+\$CRLF+a=fmtp:+\$1+" QCIF=4"+\$CRLF		

The result of manipulating the original SIP INVITE (shown above) with the configured SIP manipulation is:

```

INVITE sip:service@192.168.1.61:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.60:5060;branch=z9hG4bK-1-0
From: sipp <sip:sipp@192.168.1.60:5060>;tag=1
To: sut <sip:service@192.168.1.61:5060>
Call-ID: 1-15554@192.168.1.60
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.1.60:5060;user=phone>
Max-Forwards: 70
Content-Type: multipart/mixed;boundary=boundary
Content-Length: 466
--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=audio 12345 RTP/AVP 8
a=rtpmap:18 G729/8000/1
a=fmtp:18 annexb=no
a=sendrecv
a=maxptime:200

--boundary
Content-Type: application/sdp
v=0
o=user1 53655765 2353687637 IN IP4 192.168.1.60
s=-
c=IN IP4 192.168.1.60
t=0 0
m=video 12345 RTP/AVP 34
a=rtpmap:34 H263/90000
a=fmtp:34 QCIF=4

--boundary-

```


Local Policy Session Agent Matching

When you enable the local policy session agent matching option in your global SIP configuration, you change the way local policies match session agents. Normally, the Net-Net SBC looks up and stores matched session agents configured as next hops so it does not need to perform the lookup while processing requests. In this type of matching, the Net-Net SBC does take the realm set in the local policy attributes into consideration. When the Net-Net SBC performs its regular matching method and you have enabled overlapping IP addresses for session agents, the Net-Net SBC might match session agents to different realms than the ones you intended when creating your configuration.

Local policy session agent matching provides a way to match session agents differently, taking realms and nested realms into consideration during the matching process. This difference is key to deployments with multiple peering partners that use the overlapping IP address feature, and have multiple local policies routing to the same IP address in different realms where some target next hops require session constraints but others do not. In the cases where no session constraints are required, session agents are not needed. But session agents still match the local policy, applying their constraints, because they match the next hop IP address.

In addition to modifying this behavior, this feature also affects the use of realms and nested realms. It triggers the use not only of realms, but of all the realms nested however deeply—thereby improving matching efficiency.

How It Works

You can set the local policy session agent matching option with values that define how the Net-Net SBC performs session agent matching:

- **any**—The Net-Net SBC looks up and stores matched session agents configured as next hops so it does not need to perform the lookup while processing requests, without regard to realms.

This behavior is the default when the SIP configuration does not have the local policy session agent matching option set.

- **realm**—The Net-Net SBC selects session agents in the realm that the local policy attribute indicates; this provides an exact match, rather than not taking the realm into consideration during session agent selection.

For example, the session agent is a match if the session agent **realm-id** and the local policy attribute **realm** parameters are an exact match.

- **sub-realm**—Session agents in the same realm or the same realm lineage—where session agents and realms are related to one another via realm parent-child relationships no matter the depth of realm nesting configured

For example, the session agent is a match if the local policy attribute **realm** is a sub-realm of the realm specified in the session agent **realm-id** parameter.

- **interface**—Session agents in the same realm or same realm lineage via the realm set in the local policy attribute, and whose realm uses the same signaling interface as the realm set in the local policy attribute

For example, the session agent is a match if the session agent **realm-id** is a sub-realm of the local policy attribute **realm**, and both referenced realms use the same SIP signaling interface.

- **network**—Session agents whose realm is in the realm lineage for the same realm set in the local policy attributes, and whose realm is associated with the same network interface as the realm set in the local policy attributes

For example, the session agent is a match if the session agent **realm-id** is a sub-realm of the local policy attribute **realm**, and realm reference by both use the same network interface.

If it cannot find a match, the Net-Net SBC will use the IP address as the next hop. Further, requests matching local policy attributes will not be associated with session agents, and so their constraints will not be applied.

The Net-Net SBC stores session agent information that it looks up when performing local policing session agent matching. To perform the lookup, it uses the session agent hostname as a key. When the hostname is an FQDN and there is a configured IP address, the Net-Net SBC uses the IP address as a secondary key. Given this implementation, the following are true when selecting session agents:

- If multiple session agents share the same IP address, the one with an IP address in the hostname parameter takes precedence.
- If all session agents with the same IP address have an FQDN as their hostname, the one whose name is alphabetically lower will take precedence, where “alphabetically lower” means earlier in the alphabet (closer to A than to Z).
- For non-global session agents (whose realms are configured but not wildcarded) with an IP address, the Net-Net SBC uses a key that is a combination of the IP address and the realm in the form <address>:<realm>.
- For a session agent whose realm has a parent realm, the Net-Net SBC uses a combination of the IP address, realm, and realm-path (or lineage for the realm) in the form <address>:<realm-path>. For example, the realm path for a realm **core3** with a parent **core2**, which in turn has a parent **core** would be **core:core2:core3**.

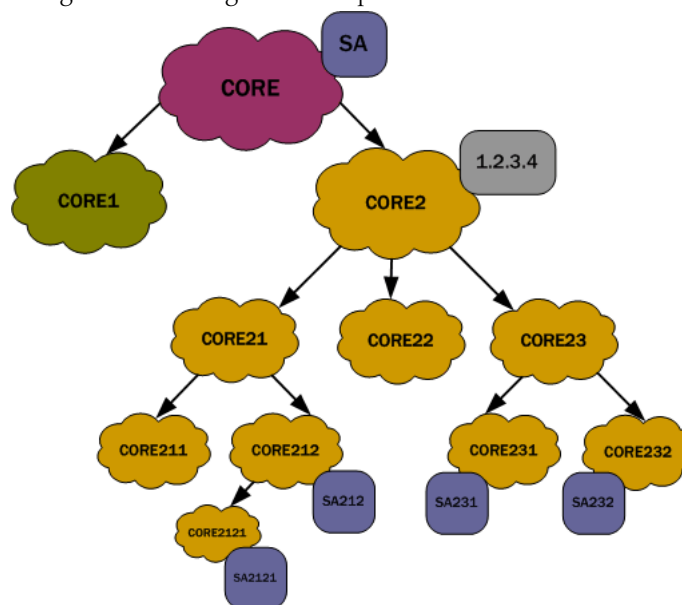
When it looks up a session agent with a realm, the Net-Net SBC first searches for an exact match for the IP address and realm combination. If this fails, it performs a second search if the desired realm has parents or children. The Net-Net SBC locates an entry in its repository of session agent information that is greater than or equal to the IP address with the base realm, which is the ancestor of the desired realm without a parent. Having gathered this set of candidates, the Net-Net SBC narrows down the search for a match by comparing sub-realms and determines there is a match if either:

- The desired realm path is a sub-string of the entry’s realm path, or
- The entry’s realm path is a substring of the desired realm path (i.e., the desired realm is a sub-realm of the entry’s realm)

Then the Net-Net SBC orders the candidates by depth of the entry’s realm-path, or number of levels from the base realm relative to the depth of the desired realm. By searching the ordered set until the entry’s realm depth equals the desired realm’s depth, the Net-Net SBC determines a parent candidate, all subsequent entries being sub-realms of the desired realm. The Net-Net SBC only considers entries at the first level deeper than the desired realm. If at this point there is only one entry, the Net-Net SBC deems it a match. Otherwise, it selects the parent candidate as the matching entry. In the event the search does not yield a matching realm, the Net-Net SBC uses the global session agent for the IP address, if there is one.

The following diagram shows the realm tree, where the clouds are realms and squares are session agents, representing a group of session agents sharing the IP

address 1.2.3.4. The Net-Net SBC searches for the session agents lower in the tree along the session agent realm-path and the desired realm.



For the diagram above, the following shows how the hostname would look for this group of session agents.

Key	Session Agent (hostname[realm])
1.2.3.4 (This session agent owns the primary key for the IP address because its hostname is the IP address.)	1.2.3.4[CORE2]
1.2.3.4:CORE (IP+realm key entry)	SA[CORE]
1.2.3.4:CORE (IP+realm key entry)	1.2.3.4[CORE2]
1.2.3.4:CORE212 (IP+realm key entry)	SA212[CORE212]
1.2.3.4:CORE2121 (IP+realm key entry)	SA2121[CORE2121]
1.2.3.4:CORE231 (IP+realm key entry)	SA231[CORE231]
1.2.3.4:CORE232 (IP+realm key entry)	SA232[CORE232]
1.2.3.4:CORE: (IP+realm-path key entry)	SA[CORE]
1.2.3.4:CORE:CORE2: (IP+realm-path key entry)	1.2.3.4[CORE2]
1.2.3.4:CORE2:CORE21:CORE212 (IP+realm-path key entry)	SA212[CORE212]

Key	Session Agent (hostname[realm])
1.2.3.4:CORE2:CORE21:CORE212:CORE2121 (IP+realm-path key entry)	SA2121[CORE2121]
1.2.3.4:CORE2:CORE23:CORE231 (IP+realm-path key entry)	SA231[CORE231]
1.2.3.4:CORE2:CORE23:CORE232 (IP+realm-path key entry)	SA232[CORE232]

For each realm in the table above, the search results for each realm would look like this:

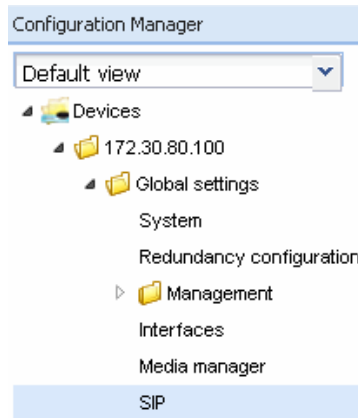
IP Address	Realm	Session Agent (hostname[realm])
1.2.3.4	CORE	SA[CORE]
1.2.3.4	CORE2	1.2.3.4[CORE2]
1.2.3.4	CORE21	SA212[CORE212]
1.2.3.4	CORE211	1.2.3.4[CORE2]
1.2.3.4	CORE212	SA212[CORE212]
1.2.3.4	CORE2121	SA2121[CORE2121]
1.2.3.4	CORE22	1.2.3.4[CORE2]
1.2.3.4	CORE23	1.2.3.4[CORE2]
1.2.3.4	CORE231	SA231[CORE231]
1.2.3.4	CORE232	SA232[CORE232]

Configuring Local Policy Session Agent Mapping

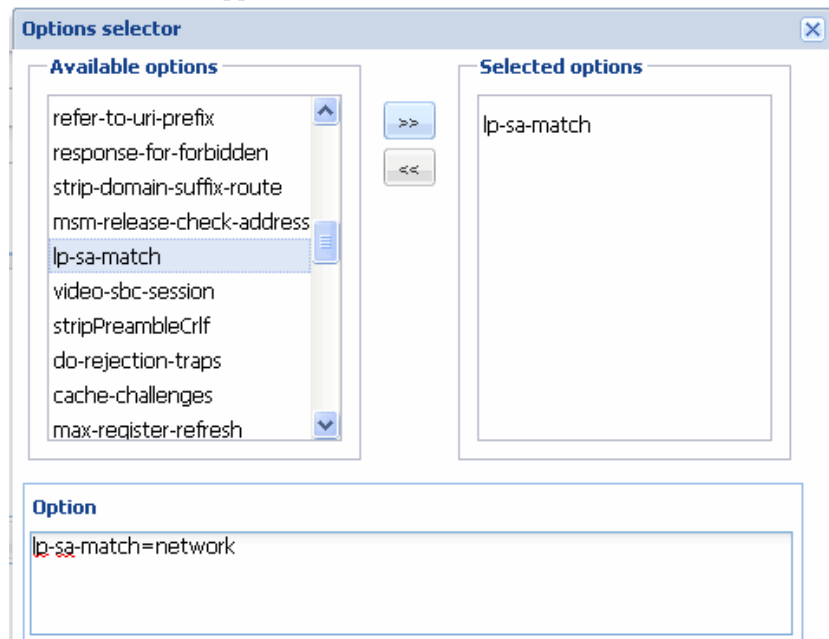
When you enable local policy session agent matching, remember that you can choose from five different ways to use the feature. See [How it Works](#) for details.

To enable local policy session agent matching using the realm method:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **Ip-sa-match** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



7. **Ip-sa-match**—Set the method for this option. The valid methods are: any, realm, sub-realm, interface, and network. For example:



If you do not specify a method, the Net-Net SBC uses the **all** method.

8. Click **OK**.
9. Click **Apply**.

1532 - SIP Method Matching and To Header Use for Local Policies

For SIP, this feature grants you greater flexibility when using local policies and has two aspects: basing local policy routing decisions on one or more SIP methods you configure and enabling the Net-Net SBC to use the TO header in REGISTER messages for routing REGISTER requests.

SIP Methods for Local Policies

This feature allows the Net-Net SBC to include SIP methods in routing decisions. If you want to use this feature, you set a list of one or more SIP methods in the local policy attributes. These are the SIP methods you can enter in the list: INVITE, REGISTER, OPTIONS, INFO, SUBSCRIBE, NOTIFY, REFER, MESSAGE, and PUBLISH.

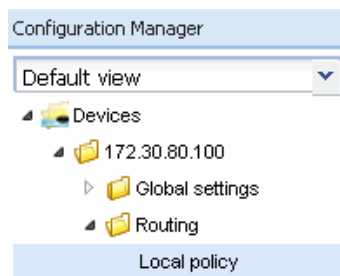
After the Net-Net SBC performs a local policy look-up for SIP, it then searches for local policy attributes that have the request method configured in the p. If it finds a set of policy attributes that matches a method that matches the traffic it is routing, the Net-Net SBC uses that set of policy attributes. This means that the Net-Net SBC considers first any policy attributes with methods configured before it considers those that do not have methods. In the absence of any policy attributes with methods, the Net-Net SBC uses the remaining ones for matching.

In cases where it finds neither matching policy attributes with methods or matching policy attributes without them, the Net-Net SBC either rejects the calls with a **404 No Routes Found** (if the request calls for a response) or drops the call.

You configure local policy matching with SIP methods in the local policy attributes parameter calls **Methods**. This parameter is a list that takes either one or multiple values. If you want to enter multiple values, you put them in the same command line entry, enclosed in quotation marks and separated by spaces.

To configure SIP methods for local policy matching:

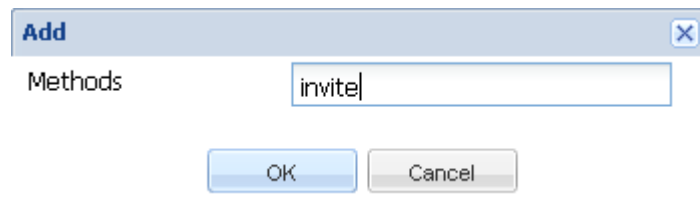
1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Select a policy name in the table and click **Edit**.
4. Scroll to the Route policy attributes table, select an attribute and click **Edit**.
5. Scroll to Methods.
6. Click **Add**. The Methods dialog box appears.

7. **Methods**—Enter the SIP method you want to use for matching this set of policy attributes. By default, this parameter is empty—meaning that SIP methods will not be taken into consideration for routing based on this set of policy attributes.

- INVITE
- REGISTER
- PRACK
- OPTIONS
- INFO
- SUBSCRIBE
- NOTIFY
- REFER
- UPDATE
- MESSAGE
- PUBLISH



8. Click **OK**.
9. Repeat steps 6 through 8 if you want to enter more than one method.
10. Click **Apply**.

Routing Using the TO Header

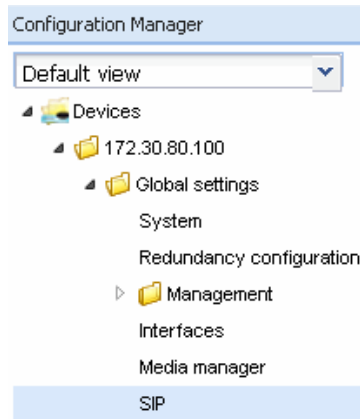
For the Net-Net SBC's global SIP configuration, you can enable the use of an ENUM query to return the SIP URI of the Registrar for a SIP REGISTER message. Without this feature enabled, the Net-Net SBC uses the REQUEST URI. This ability can be helpful because REGISTER messages only have the domain in the REQUEST URI, whereas the SIP URI in the To header contains the user's identity.

There are two parts to enabling this feature. First, you must enable the **Use To: header for local policy match on register** parameter in the global SIP configuration. Then you can set the next-hop in the applicable local policy attributes set to **ENUM**.

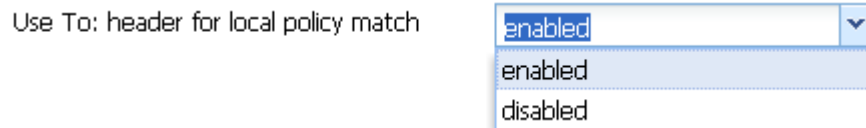
Configuring TO Header Use for Local Routing

To configure TO header use for local routing:

1. Expand the Configuration Manager slider > Global settings.



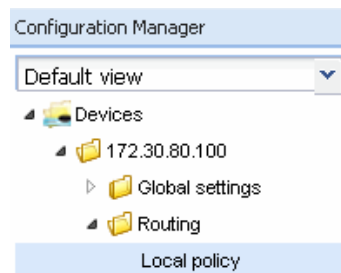
2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to Use To: header for local policy match on register.
5. **Use To: header for local policy match on register**—Select enabled from the combination box to enable this parameter if you want the Net-Net SBC to use an ENUM query to return the SIP URI of the Registrar for a SIP REGISTER message. This parameter defaults to **disabled**.



6. Click **Apply**.

To set up your local policy attributes for routing using the TO header:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Select a policy name in the table and click **Edit**.
4. Scroll to the Route policy attributes table, select an attribute and click **Edit**.
5. **next-hop**—This is the next signaling host. Set this parameter to ENUM if you want to use SIP methods in local policy attribute information for routing purposes.

6. Click **Apply**.

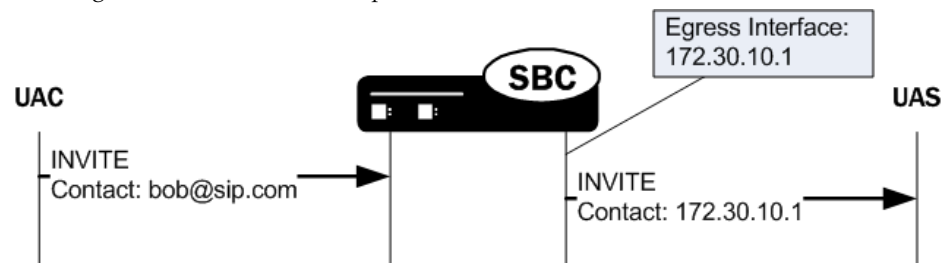
SIP Globally Routable User Agent URIs

SIP Globally Routable User Agent (UA) URIs (GRUU) are designed to reliably route a SIP message to a specific device or end user. This contrasts with a SIP AoR which can refer to multiple UAs for a single user, thus contributing to routing confusion. The Net-Net SBC can perform different behaviors when it finds SIP GRUUs in Contact headers.

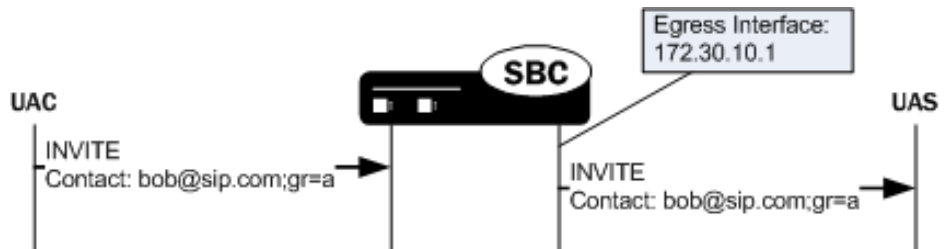
User agents supporting GRUU include a GRUU-identifying parameter in the Contact header of a dialog forming and target refresh requests. The Net-Net SBC scans for the GRUU parameter in the Contact header only when the endpoint it receives a request from is registered or when the **Pass through GRUU contact** parameter is enabled.

Contact Header URI Replacement

When no GRUU is encountered in the contact header, and when a SIP message is forwarded to the egress realm, the contact header's URI is replaced with the Net-Net SBC's egress interface. For example:



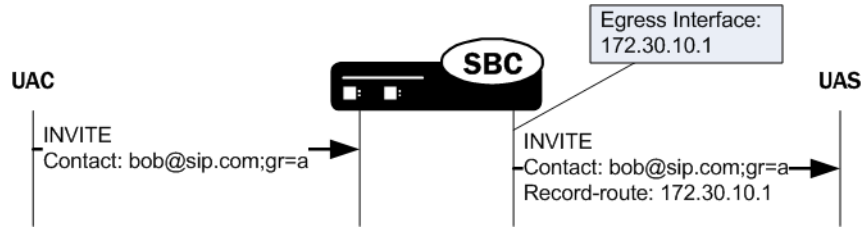
When the Net-Net SBC forwards a request where the original Contact header contains a GRUU, the contact header's URI is forwarded unchanged on the egress side of the call. For example:



Record-Route Addition

When the request is forwarded to a realm where the endpoint's registrar does not exist, the Net-Net SBC adds a Record-Route header containing the egress SIP

interface address. This causes subsequent replies or requests addressed to the GRUU to be routed through the Net-Net SBC first.



When the request is forwarded to the realm where the registrar exists, adding the Record-Route header is unnecessary and does not occur. This is because subsequent requests are directed to the registrar which will ultimately forward them to the Net-Net SBC using the registered Contact in the Request-URI.

GRUU URI Parameter Name

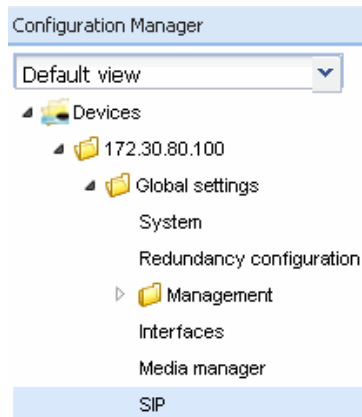
The Net-Net SBC scans for a **gr** URI parameter in the contact header to identify it as a GRUU as defined in the ietf draft[2]. The Net-Net SBC can be configured to scan for a **gruu** URI parameter in the contact header too. This alternate behavior is enabled with the **scan-for-ms-gruu** option and is used to interact with the Microsoft Office Communications Server unified communications product. When **scan-for-ms-gruu** is enabled, the Net-Net SBC scans first for the **gruu** URI parameter. If not found, it then scans for **gr** URI parameter.

Configuring SIP GRUU

This section shows you how to configure the GRUU support for non-registered contacts. Enabling GRUU functionality to parse for **gr** URI parameter rather than the IETF standard **gruu** parameter is also provided.

To configure SIP GRUU:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click the Registration tab at the top of the content area.
5. Scroll to **Pass through GRUU contact**.

6. **Pass through GRUU contact**—Select enabled from the drop-down list to activate parsing for **gr** URI parameter in the contact header in non-registered endpoints' messages and then pass the messages through the system.

Pass through GRUU contact

7. Click **Apply**.
8. Click the **System** tab at the top of the content area.
9. Click **Options**. The Options dialog box appears.
10. Click **scan-for-ms-gruu** in the Available options list to select it.
11. Click  to move the option to the Selected options list. The option name appears along with a description in the description section of the window.

12. Click **OK**.
13. Click **Apply**.

Whitelists

1623

The Session Director (SD) on the Net-Net Session Border Controller (SBC) currently ignores and passes-through unknown SIP headers and URI parameters. However, some operators require that the SD only accept messages with headers and URI parameters complying with those supported by their internal equipment. This section describes the use of whitelists to control unknown headers and parameters in request and response traffic.

What is a Whitelist?

A whitelist is an approved list of entities for which equipment provides particular privileges, access, and recognition. The SD can use configured whitelist profiles to control and accept specific inbound SIP headers and URI parameters that are being passed-through the Net-Net SBC. When you configure this feature, the SD rejects requests not matching the configured profile, or removes the unspecified headers or URI parameters not in the configured profile.

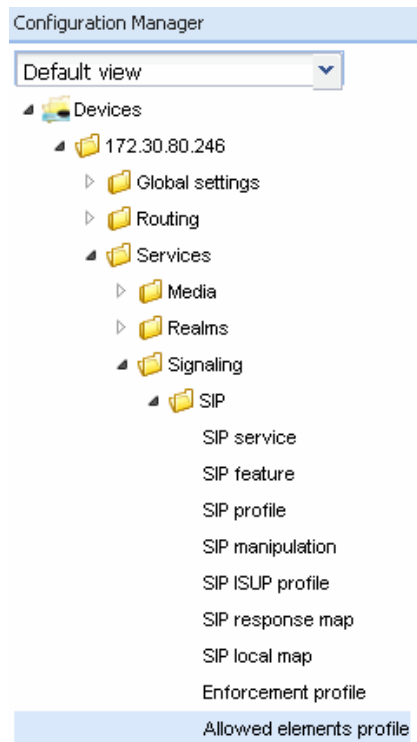
Configuring Whitelists

You can configure whitelist profiles or rules that allow the Net-Net SBC to only accept inbound SIP headers and URI parameters that are configured in this whitelist, using the parameter **Allowed elements profile**. Since the **Enforcement profile** value also pertains to session agents, realms, and SIP interfaces, you can also apply the profiles you configure to these remote entities.

In the following configuration example, it is assumed that your baseline configuration passes SIP traffic, with the Net-Net SBC in the role of an Access SBC. Use this procedure to configure a whitelist for the session router and optionally apply the specific whitelists to the session agent and SIP interface, as well as the media manager's realm configuration.

To configure a whitelist for the session router:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click Allowed elements profile. The Allowed elements profile table displays in the content area.
3. ***Name** — Enter a unique name for the whitelist you are creating. This name can be referenced when configuring the enforcement-profiles for session agent, SIP interface, and realms.

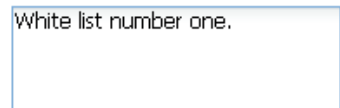
Add Allowed elements profile

*Name

4. Click **Apply**.
5. Click the white list you want to configure in the table and click **Edit**.

6. **description** — Enter a description that explains the purpose of creating this whitelist. You can use any alpha-numeric characters when entering the description.

Description


 White list number one.

7. **Rule sets**—Click **Add**. The Add Rule sets dialog box appears.
8. **Name**—Enter a unique name for is rule set. This field is required.
9. **Logging** — Click **enabled** in the drop down list so an incoming message is written to a log file called “*matched.log*” when the message contains an element not specified in the whitelist. The default value is **disabled**. The valid values are: enabled | disabled.
10. **Message type** — Specify the type of messages for which the Net-Net SBC applies this whitelist configuration. The default value is **any**. The valid values are:
- **any** — Applies to all incoming messages.
 - **request** — Applies to only incoming REQUEST messages.
 - **response** — Applies to only incoming RESPONSE messages.
11. **Unmatched action** — Select the action for the Net-Net SBC to perform when a header does not exist in an incoming message. The default value is **reject**. The valid values are:
- **reject** — Rejects all incoming messages that do not contain a header.
 - **delete** — Deletes all incoming message that do not contain a header.

Note: This parameter applies to non-matching header names only (not non-matching URI parameters).

*Logging	enabled ▼	(Default:disabled)
*Message type	any ▼	(Default:any)
*Unmatched action	reject ▼	(Default:reject)
Apply Cancel		

12. Click **Apply**. The rule set is added to the table.
13. Click the rule set you want to configure in the table and click **Edit**.
14. **Methods** —Click **Add**.
15. **Methods**—Enter the packet method(s), separated by a comma, for which this whitelist is enforced. Packet methods include, INVITE, OPTIONS, ACK, BYE, etc. If this field is left blank, the whitelist applies to all packet methods. You can enter up to a maximum of 255 characters.
16. Click **OK**. The method is added to the Methods table.
17. Repeat steps 15 and 16 to add additional methods.

About the Matched.log

The *matched.log* contains information about the timestamp, received/sent SD network-interface, IP address/port from which it was received or being sent from, and which peer IP address/port it was received from or sent to. The log also specifies

the request URI (if applicable), and the From, To, and Contact headers in the message, as well as which rule triggered the log action. An example of the log output of the *matched.log* file is as follows:

```
Dec 17 14:17:54.526 On [0:0]192.168.1.84:5060 sent to 192.168.1.60:5060
allowed-elements-profile[whitelist1(reject)]
INVITE sip:service@192.168.1.84:5060 SIP/2.0
From: sipp <sip:+2125551212@192.168.1.60:5060>;tag=3035SIPpTag001
To: sut <sip:service@192.168.1.84>
Contact: sip:sipp@192.168.1.60:5060
```

The **Header rule** object consists of 6 parameters that make up the header-rule:

- Name
- Unmatched action
- Allowed header parameters
- Allowed uri parameters
- Allowed uri user parameters
- Allowed uri header names

You can configure an unlimited number of header-rules on the Net-Net SBC that you can apply to your network. Use the following parameters to configure a header-rule that the Net-Net SBC uses to control which incoming messages it allows.

The Header rule object allows you to configure, as part of the whitelist, multiple parameters which make up the header rule that the SD allows from incoming messages. Header-rules do NOT have to be in any specific order. The system prompt changes to let you know that you can begin configuring individual parameters for this object.

Configuring Header Rules

To configure header rules:

1. From the Allowed elements profile window, click **Add** under Header rules. The Add Header rules dialog box appears.
2. **Name**—Enter the name of the header in the whitelist that the Net-Net SBC allows from incoming messages. It is case-insensitive and supports abbreviated forms of header names. For example, “Via”, “via”, or “v” all match against the same header. A header name of “request-uri” refers to the request URI of requests, while a header name of “*” applies to any header-type not matched by any other header-rule. The default value is “*”. This default value provides the ability to have header-rules for commonly known headers that remove unknown parameters, but leave unknown headers alone.
3. **Unmatched action**—Select the action for the Net-Net SBC to perform when an incoming header’s parameters do not match the relevant allowed parameters specified for this header-name. The default value is **reject**. The valid values are:
 - **reject** — Rejects all incoming messages that have header parameters that do not match the parameters specified in this header-name.
 - **delete** — Deletes all incoming messages that have header parameters that do not match the parameters specified in this header-name.

Note: This parameter applies to non-matching header names only (not non-matching URI parameters).

4. **Allow header parameters**—Enter the header parameter that the Net-Net SBC allows from the headers in incoming messages. You can enter up to 255 characters, including a comma (,), semi-colon (;), equal sign (=), question mark (?), at-symbol (@), backslash (\), or plus sign (+). The default value is "*", which allows all header parameters to pass through. If you leave this field empty, no header parameters are allowed.
5. **Allow uri parameters**—Enter the URI parameter that the Net-Net SBC allows from the headers in incoming messages. You can enter up to 255 characters, including a comma (,), semi-colon (;), equal sign (=), question mark (?), at-symbol (@), backslash (\), or plus sign (+). The default value is "*", which allows all URI parameters to pass through. If you leave this field empty, no URI parameters are allowed.
6. **Allowed uri user parameters**—Enter the URI user parameter that the Net-Net SBC allows from the headers in incoming messages. You can enter up to 255 characters, including a comma (,), semi-colon (;), equal sign (=), question mark (?), at-symbol (@), backslash (\), or plus sign (+). The default value is "*", which allows all URI user parameters to pass through. If you leave this field empty, no URI user parameters are allowed.
7. **Allowed uri header names**—Enter the URI header name that the Net-Net SBC allows from the headers in incoming messages. You can enter up to 255 characters, including a comma (,), semi-colon (;), equal sign (=), question mark (?), at-symbol (@), backslash (\), or plus sign (+). The default value is "*", which allows all URI header name parameters to pass through. If you leave this field empty, no URI header name parameters are allowed.

Add Header rules
✕

*Name	*	(Unique required)
Sequence number	2	
*Allowed uri user parameters	*	(Default:*)
*Allowed uri parameters	*	(Default:*)
*Allowed header parameters	*	(Default:*)
*Unmatched action	reject ▼	(Default:reject)
*Allowed uri header names	*	(Default:*)

8. Click **Apply**.

Configuration Exception

There are specific instances in incoming Request-URI messages where the SD ignores specific parameters and automatically adds header-rules.

In a Request-URI, all parameters are URI parameters, and URI headers are not allowed. If you define values for the “allow-header-param”, “allow-uri-header-name”, and “allow-uri-param”, the SD ignores these parameters in the Request-URI. Instead the SD automatically adds header-rules for incoming “Via”, “From”, “To”, “Call-ID”, and “CSeq” messages. These are explicit header rules and cannot be deleted. Each header-rule in a Request-URI has parameters populated with the value of “*”. If required, a user can change the header-rule parameter values with the values identified in the following table.

Header Rule	Applicable Parameter	Required Value(s)
Via	allow-header-param	• branch • received • rport
From	allow-header-param	• tag
To	allow-header-param	• tag
Call-ID	allow-header-param	No restrictions
CSeq	allow-header-param	No restrictions

How Whitelists Work

Whitelists allow you to customize which SIP signaling messages to allow into your network and which messages to reject or delete. In the flow of SIP traffic to/from the Net-Net SBC, the SD matches any received request or response, in or out of a dialog against the configured allowed list, and rejects or deletes the non-matching element based on the actions specified in the whitelist configuration.

For responses, the SD does not reject the message if a header or parameter is not found in the allowed list, even if the action is set to reject. Instead it deletes the offending parameter or header. In addition, if the message is a request of the method type ACK, PRACK, CANCEL or BYE, it deletes all unmatched elements, but does not reject the request, even if the action was configured to reject.

The whitelist verification performs for any method; however you can narrow this list to operate only on specific methods by defining them in the **methods** parameter of the configuration.

Whitelist verification occurs when a request or response is received by the SD, but only after the SD has processed the inbound header manipulation rule (HMR), network management controls (NMC), Resource-Priority header (RPH), and monthly-minutes checking.

The SD responds to requests which have non-matching headers or parameters configured with an action of “reject”, with a “403 Forbidden” response by default. You can use a local-response event, **allowed elements profile rejection**, to override the default reject status code and reason phrase.

The configured whitelist operates transparently on headers that have multiple URIs or multiple header values within a single header (header values separated by a comma).

Parameter parsing operates only on parameters that it can identify. For parameters that can not be parsed, for example an invalid URI (e.g. < sip:user@host.com&hp=val>), the SD ignores this URI header parameter value of "hp" since it is not contained within a valid URI. Even though it would appear to be a URI header parameter, URI headers must come after URI parameters. Parameter matching does not occur if the headers and parameters in the URI are not well-formed. The SD does not remove the parameter since it cannot identify it.

Whitelist SNMP Traps

The Management Information Base (MIB) contains information about SNMP traps, which enable an agent to notify the management station of significant events by way of an unsolicited SNMP message. When an element sends a TRAP packet, it can include OID and value information (bindings) to clarify the event.

The Net-Net SBC has two Simple Network Management Protocol (SNMP) traps that exist within the session-router object that are triggered when counter limits are exceeded during incoming traffic managed by the whitelist. These SNMP traps are as follows:

SNMP Trap	Description
reject-message-window	Generated when the SD rejects messages due to matching criteria in the whitelist. This trap specifies the number of messages rejected.
reject-message-threshold	Generated when the number of rejected messages exceeds the configured threshold within the configured window.

The global counter that resides on the SD is used to track messages that are flagged for rejection during use of the whitelists. This is the same counter shared with the sip-manipulation object, "reject" action.

Once the total of rejected messages exceeds the configured threshold for a period, the traps are sent and no more traps get generated until another period's counter exceeds the configured limit. The value of 0 is disabled. The default value is 0.

When the whitelist feature is used in conjunction with the HMR reject message feature, both features increment the same counters and trigger the same SNMP trap. The trap specifies that it was triggered only, and does not indicate which feature triggered the trap. However, the trap information in the *matched.log* file indicates which messages and what rules caused the trap to occur.

The following MIB objects support the whitelist SNMP traps:

Object	Supported MIB	Description
apSysRejected Messages (read-only)	apSysMgmtMIBGeneralObjects 18	Number of messages rejected by the SD due to matching criteria.
apSysMgmtRejectedMesages ThresholdExeededTrap	apSystemManagementMonitors 57	Generates when the number of rejected messages exceeds the configured threshold within the configured window.

Object	Supported MIB	Description
apSysMgmtRejectedMessage sGroup	apSystemManagementGroups 18	Objects to track the number of messages rejected by the SD.
apSysMgmtRejectedMessage sNotificationsGroup	apSystemManagementNotifications Groups 26	Traps used for notification of rejected messages.
apSmsgmtRejectedMessagesC ap	apSmsgmtMibCapabilities 37 (includes apSysMgmtRejectedMessagesGrou p, apSysMgmtRejectedMessagesNotifi cationsGroup)	Acme Packet Agent Capability for enterprise

Whitelist Learning

You can build your whitelist configuration based on the “learning” capabilities of the SD on the Net-Net SBC. When you enable the SD “learning” mode, it acquires the knowledge of the allowable elements (headers and parameters) currently incoming to your network. The SD collects the information about the headers received and the parameters that exist within each header. The information continues to be gathered until you disable the “learning” mode.

Once you disable the “learning” mode, the SD prompts you to enter a name for the allowed-elements-profile. If the profile name you entered does not exist, the captured information is written to the new allowed-elements-profile configuration. The administrator can then make changes to the configuration as applicable, save the configuration, and apply it to a logical remote entity.

The new allowed-elements-profile does not contain any wildcard rules. The SD cannot generate wildcard headers and parameters during the “learning” mode. The Methods object is populated from the list of methods seen by the SD while learning.

Note: Acme Packet recommends running the “learning” mode during off-peak and/or light traffic times. This mode can operate in conjunction with the execution of an allowed-elements-profile. The learning occurs just before any configured allowed-elements-profile configuration.

Start/Stop Whitelist Learning

The ACLI interface provides two commands that allow a Superuser to start and stop whitelist learning on the Net-Net SBC:

Command	Description
start <argument> <options>	Starts whitelist learning on the Net-Net SBC. - You must specify the argument learn-allowed-elements with this command to start the learning operation. - Optionally, you can use method, msg-type, and params after the argument.
stop <argument> <identifier>	Stops the whitelist learning on the Net-Net SBC and writes the learned configuration to the editing configuration on the SD where it is saved and activated. - You must specify the argument learn-allowed-elements with this command to stop the learning operation. - You must specify a unique identifier that identifies the allowed-elements-profile name. Note: If you specify an identifier name that already exists as a profile, the ACLI returns an error message and prompts you to enter a different name.

You can use these commands at the top level ACLI prompt as required on the Net-Net SBC.

You use these commands with the argument, “**learn-allowed-elements**” to start/stop the whitelist learning feature. By default, the learning mode creates a single rule-set under which all of the headers and their respective parameters are stored.

For example:

```
ACMEPACKET# start learn-allowed-elements
Learning mode for allowed-elements-profile started.
```

In the above example, “**start**” is the top level ACLI command and “**learn-allowed-elements**” is the operation being performed.

Optionally, you can specify [**method**], [**msg-type**], and [**params**] in any order, for the SD to learn specific rule-set elements from incoming messages and save them to the whitelist configuration.

For example:

```
ACMEPACKET# start learn-allowed-elements method msg-type params
```

The “**method**” option creates a new rule-set per unique method. The “**msg-type**” option creates a new rule-set per unique message-type seen. The “**params**” option performs URI and header parsing to examine parameters within the message. By default, parameter parsing is disabled.

To start the whitelist learning feature:

In Superuser mode, at the top level ACLI prompt, type **start learn-allowed-elements** and press <Enter>.

```
ACMEPACKET# start learn-allowed-elements
```

The following message displays:

```
Learning mode for allowed-elements-profile started.
```

To specify the elements of rule-sets for whitelists:

In Superuser mode, at the top level ACLI prompt, type **start learn-allowed-elements method msg-type params** and press <Enter>.

ACMEPACKET# **start learn-allowed-elements method msg-type params**

The following message displays:

Learning mode for allowed-elements-profile started.

Note: If you try to start a whitelist learning operation while another learning operation is already running, the following message displays:

Learning mode restarted without saving

Learning mode for allowed-elements-profile started.

To stop the whitelist learning feature:

In Superuser mode, at the top level ACLI prompt, type **stop learn-allowed-elements <identifier>**, where <identifier> is the allowed-elements-profile name, and press <Enter>.

ACMEPACKET# **stop learn-allowed-elements whitelist1**

The following message displays:

Learning mode for allowed-elements-profile stopped.

Note: If you specify an identifier name that already exists as a profile, the ACLI returns an error message and prompts you to enter a different name.

Displaying Rejected Messages

Whitelists, when configured on the Net-Net SBC, control whether or not the SD allows unknown headers and URI parameters to be accepted in incoming request and response traffic. When the SD rejects messages according to the whitelist, the rejected messages are logged to a file called *“matched.log”* if logging is set to **“enabled”**. You can open and view the log when required to view the rejected messages.

In addition to the rejected messages being logged to the *“matched.log”* file, the rejected messages are also sent through a burst counter that keeps track of the amount of messages rejected. On the SD4, you can display the number of rejected messages by entering the **“show sip transport”** command as shown below. On the SD2/3, you can enter the **“show sipd”** to display the number of rejected messages.

```

Mantra# show sip transport
19:23:27-199 sip@s0.4.1
SIP Transport

```

	Active	Recent High	Total	Lifetime Total	PerMax	High
Server Trans	0	2	4	19	3	3
Client Trans	0	1	2	17	3	2
Context IDs	0	2	4	19	3	3
Sockets	0	0	0	0	0	0
Inbound Connections	0	0	0	0	0	0
Outbound Connections	0	0	0	0	0	0
Connection Aliases	0	0	0	0	0	0
Pending Messages	0	0	0	0	0	0
Connections Rejected	-	-	0	0	0	
Connection Errors	-	-	0	0	0	
Pending Msg Dropped	-	-	0	0	0	
Reg Cache Updates	-	-	0	0	0	
Overload Rejects	-	-	0	0	0	
Suppressed Retransmit	-	-	0	0	0	
Response Retransmit	-	-	20	50	10	
Request Dropped	-	-	0	0	0	
Response Dropped	-	-	0	0	0	
Invalid Requests	-	-	0	0	0	
Invalid Responses	-	-	0	0	0	
Invalid Messages	-	-	0	0	0	
Transaction Timeouts	-	-	0	0	0	
Duplicate Transactions	-	-	0	0	0	
Transaction Errors	-	-	0	0	0	
Application Errors	-	-	0	0	0	
Redundancy Errors	-	-	0	0	0	
Rejected Message(10s)	0	1	2	5	1	1

Conditional Bandwidth CAC for Media Release

The Net-Net SBC supports conditional call admission control (CAC) using the SIP profile configuration. You can allow the conditional admission of SIP calls that could potentially have their media released, instead of risking the possible rejection of those calls due to internal bandwidth limits.

About Conditional Bandwidth CAC for Media Release

The Net-Net SBC performs bandwidth CAC for SIP per realm, for each Address of Record (AoR) or IP address. The system checks bandwidth limits based on the codecs listed in SDP. If a new SIP INVITE contains codecs in an SDP message that exceed bandwidth available for a given resource, the system rejects that INVITE. This check occurs both on the ingress and egress sides of a call, and both sides must have enough available resources to support the call for it to be admitted.

In the case of calls where media is released, the Net-Net SBC does not count bandwidth consumed by the call. However, this exemption is not given until the media is actually released—and media release conditions are unknown at the time SIP INVITE is admitted. This is because an INVITE received on one side of the Net-Net SBC is only media-released when that INVITE is routed back through the Net-Net SBC as a hairpin or other multi-system media release. So there has to be enough bandwidth for the initial INVITE; otherwise, and even if the INVITE is a candidate for media release, it will be rejected.

When there is a significant volume of such calls—ones that are candidates for media release, but cannot be admitted because of CAC limits—it becomes important to admit them so long as they truly end in media release. This feature thus allows admission of SIP calls that might otherwise be rejected because of bandwidth limitations when the far-end of the call causes media to be released.

Details and Conditions

This feature applies in a two system scenario. In order to track a call as a candidate for provisional media release, the access-side Net-Net SBC adds a Require: header with an option tag to the INVITE or UPDATE message on egress. The option tag is configurable in the SIP configuration. The default is *com.acmepacket.cac*.

The following sections describe when the SIP INVITE or SIP UPDATE are:

- initially received by the Net-Net SBC
- received by the second Net-Net SBC

INVITEs/UPDATEs Initially Received By Net-Net SBC

When the Net-Net SBC first receives an INVITE or UPDATE message, it considers if it should be admitted provisionally or rejected outright due to CAC bandwidth constraints. If the INVITE or UPDATE is admitted provisionally, a Require: header is inserted on egress from the system.

The Net-Net SBC inserts the Require header on egress under these conditions:

- It receives an INVITE / UPDATE with no or a non-matching Require header.
- The **Egress conditional CAC admit state** parameter in the SIP profile on the egress realm, SIP interface, session agent is set to enabled in the egress realm
- The request would otherwise be rejected because of current bandwidth CAC limits in the ingress OR egress realms
- The call is a candidate for media release in the ingress realm

A call is considered a candidate for media release when the ingress realm has any of these parameters set to disabled:

- **Within this realm serviced by this Net-Net system only**
- **Within different realms in the same subnet serviced by this Net-Net system only**
- **Media released behind same IP**
- **Release media across network interfaces**

INVITEs/UPDATEs Received by Second Net-Net SBC

The second Net-Net SBC receives the INVITE or UPDATE with the newly inserted Require: header. Standard SIP convention indicates that if the UAS receiving the request does not know how to handle the Require header, the request should be rejected.

When the following three conditions are met, the INVITE is permitted into the system for processing:

- The **Ingress conditional CAC admit state** in the SIP profile on the ingress realm, SIP interface, session agent parameter is set to enabled
- The **con-cac-tag** sip config option is configured to the same value as the received Require header's option tag
- The call is a candidate for media-release

The call is considered a candidate for media-release on the second system (indicated by the **Ingress conditional CAC admit state** parameter is set to enabled) when either the ingress or egress realms have any of these parameters set to disabled:

- Within this realm serviced by this Net-Net system only
- Within different realms in the same subnet serviced by this Net-Net system only
- Media released behind same IP
- Release media across network interfaces

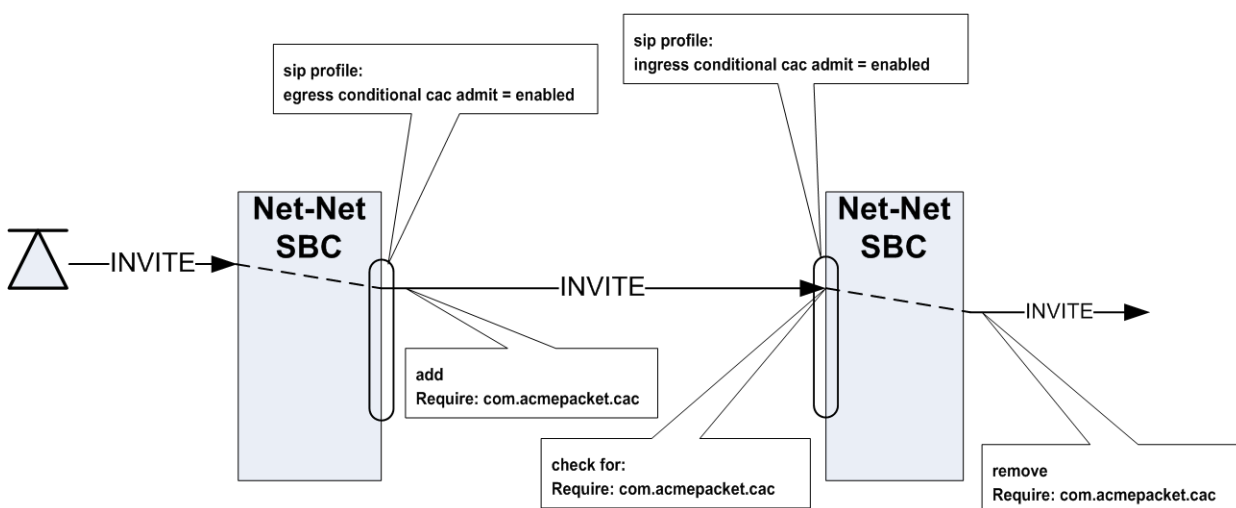
and the following parameter is set to enabled:

- Within this realm serviced by different Net-Net systems (SIP only)

If the call, as received by the second system is not considered a candidate for release, the INVITE or UPDATE is failed with a 503 “Insufficient Bandwidth” message.

After the INVITE has been processed by the Net-Net SBC, the Require: header is removed upon egress from the system.

The following diagram shows the two-system scenario:



Conditional Admission with Per-user CAC

In the event that the per-user CAC feature is also being used, and per-user CAC bandwidth is exceeded, the Net-Net SBC also uses this option tag mechanism. However, if the per-user CAC implementation does count bandwidth regardless of media-release, then the Net-Net SBC will reject calls exceeding the per-user CAC limits when it receives them.

On the second system, when the per-user CAC feature is being used, the Net-Net SBC will perform the same option tag mechanism based on if the **Ingress conditional CAC admit state** parameter is enabled.

Configuring a SIP Profile

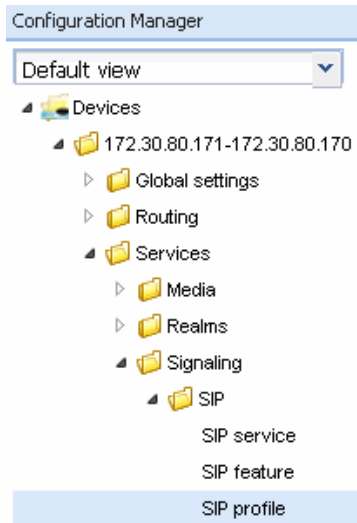
You enable this feature by first configuring a SIP profile, and then applying the profile to any of these:

- realm
- SIP interface
- SIP session agent

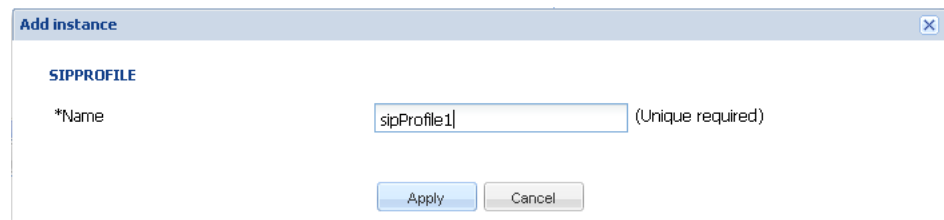
You can configure multiple SIP profiles. Though this configuration contains additional parameters, you do not have to use them for the conditional bandwidth CAC for media release.

To configure a SIP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.
3. Click **Add**. The Add SIP profile dialog box appears.
4. **Name**—Enter the name of the SIP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.



5. Click **Apply**. The new SIP profile appears in the SIP profile table.
6. Click the SIP profile in the table you want to configure and click **Edit**.
7. **Conditional CAC for media release on ingress**—Click **enabled** in the combination box to process an INVITE with a Require tag as received on an ingress interface. You can set this parameter to disabled if you do not want to use this feature on the ingress side. There is no default for this parameter.
8. **Conditional CAC for media release on egress**—Click **enabled** in the combination box if you want to use conditional bandwidth CAC for media release for calls that are first received by this system. This results in option tags being inserted on the INVITE's egress if the conditional CAC conditions are met.

You can set this parameter to disabled if you do not want to use this feature. There is no default for this parameter.

CAC

Conditional CAC for media release on ingress

enabled

Conditional CAC for media release on egress

enabled
enabled
inherit
disabled

- Click **Apply**.

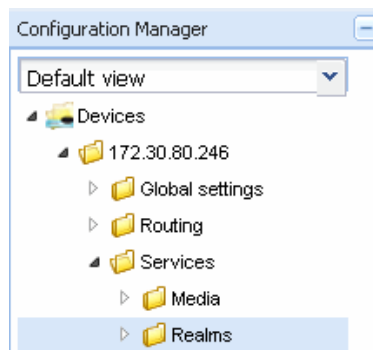
Applying a SIP Profile

Once you have configured one or more SIP profiles, you can apply them to realms, SIP interfaces, and SIP session agents.

Realms

To apply a SIP profile to a realm:

- Expand the Configuration Manager slider > Services.



- Click Realms. The Realm table appears in the content area.
- SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

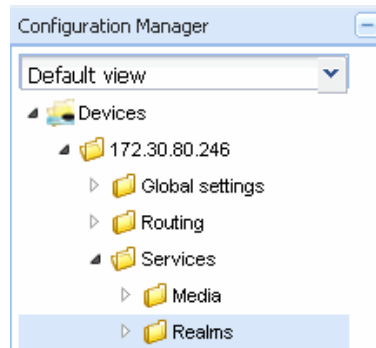
SIPprofile1

- Click **Apply**.

SIP Interface

To apply a SIP profile to a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**. The Edit Realm window appears.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

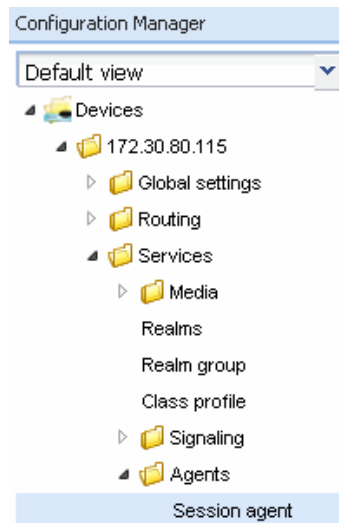
SIPprofile1

7. Click **Apply**.

Session Agent

To apply a SIP profile to a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.

3. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

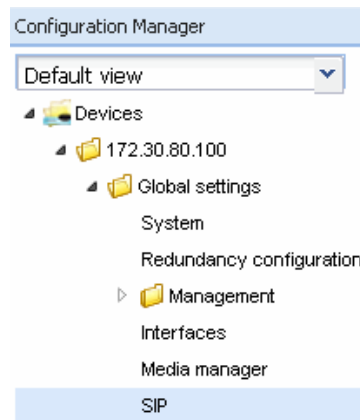
4. Click **Apply**.

Configuring Require Header Option Tag

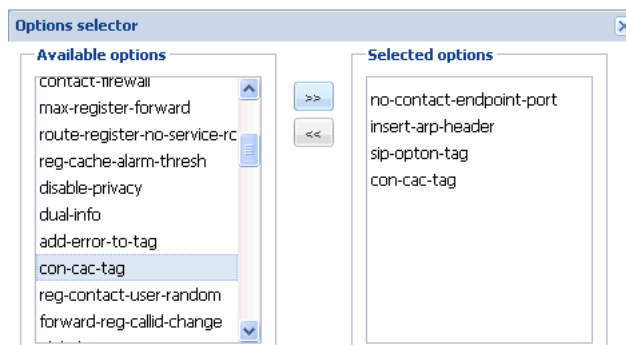
You may change the Require: header's option tag from the default `com.acmepacket.cac` to one of your own choosing. Remember that both systems' option tags must match exactly.

To configure the Require: header's option tag:

1. Expand the Configuration Manager slider > Global settings.



2. Click **SIP**.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The SIP options window appears.
5. Click **con-cac-tag** in the Available options list to select it.



6. Click **>>** to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. **con-cac-tag**—Edit the Option text box to append this option with a new tag.

8. Click **OK**.
9. Click **Apply**.

Full Via Header Replacement

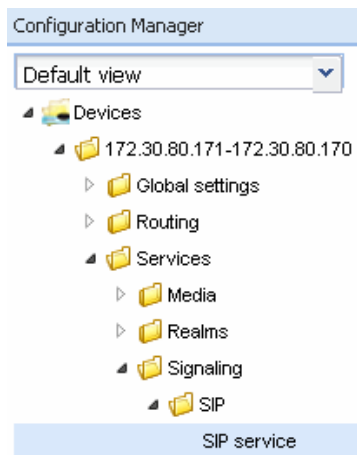
In the basic behavior of the SIP VIA Header Transparency feature, the Net-Net SBC retains the top-most VIA header received on ingress and places it under the VIA header containing the system's address on egress. All VIA headers beneath the top-most header in the ingress message are lost.

The Net-Net SBC can also retain all VIA headers received on the message's ingress. Thus the top-most VIA header becomes the second VIA header on egress, the second VIA header becomes the third VIA header on egress, and so on.

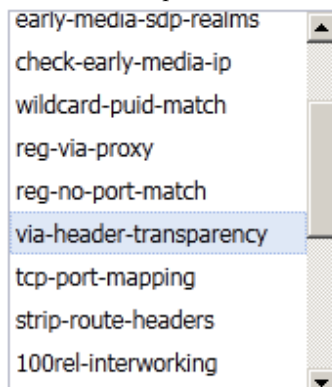
Configure SIP Via Header Transparency

To configure SIP Via header transparency for all received VIA headers:

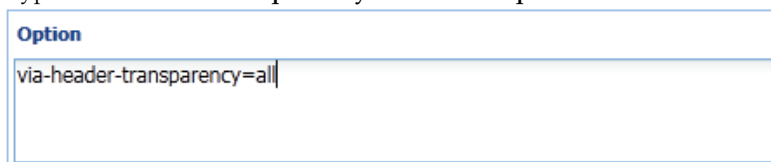
1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Select the SIP Interface you would like to use and click **Edit**.
4. Click **Options**. The Options window appears.
5. Select **via-header-transparency** in the Available Options list and click  to move the option to the Selected Options list.



6. Type **via-header-transparency=all** in the **Option** field.



The screenshot shows a configuration window with a blue border. At the top, the word "Option" is written in blue. Below it is a text input field containing the text "via-header-transparency=all". The cursor is at the end of the text in the field.

7. Click **Ok**.
8. Click **Apply**.

SIP-ISUP Format Version Interworking

ISUP messages can be carried in SIP messages through either a standard body or through a multipart MIME encoded body. While ANSI and ITU are the two major groups, each contains many specific variants. To facilitate instances where two sides of a call use different versions, the Net-Net SBC supports interworking between the following SIP ISUP formats: ANSI, ITU, ETSI-356 (an ITU variant), and GR-317 (an ANSI variant). To do so, the Net-Net SBC can move, delete, and add parameters to various sections of the message.

The ISUP message version is determined by the content type of the SIP message or the MIME content type. When the base and version parameters do not match, the Net-Net SBC first uses the base parameter value to determine the format. If there is no base, the Net-Net SBC then checks the version parameter. And if there is neither, the Net-Net SBC uses the **ISUP version** configured in the **SIP ISUP profile** configuration from the inbound realm, session agent, or SIP interface. Available values for that parameter are **ansi-2000**, **itu-99**, **gr-317**, or **etsi-356**. The Net-Net SBC considers unknown any value for the version that fails to match one of these or is missing.

Messages that contain an unknown ISUP format pass through the Net-Net SBC untouched. If there are operations to be performed on them, however, SIP ISUP HMR will take place. After the body has been converted, the Net-Net SBC updates both the base and version parameters of the content-type.

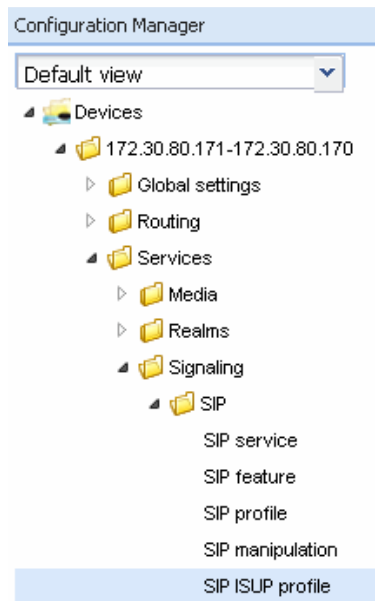
Custom formats are not supported.

Configuring SIP-ISUP Format Version Interworking

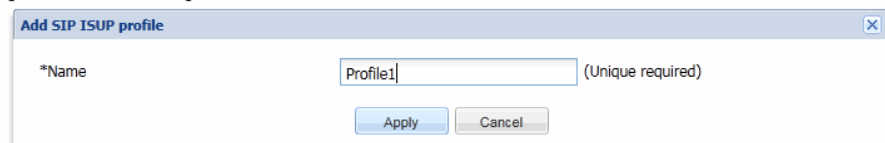
This section shows you how to set up a SIP ISUP format interworking. First, you configure a SIP ISUP profile, and then you apply it to a realm, session agent, or SIP interface.

To configure SIP-ISUP format version interworking:

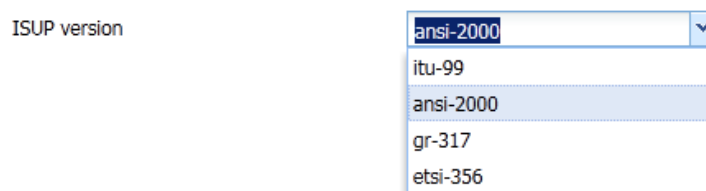
1. Expand the Configuration Manager slider > Services > Signaling > SIP.



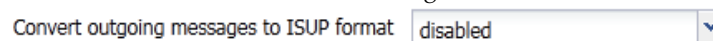
- Click SIP ISUP profile. The SIP ISUP profile table appears in the content area.
- Click **Add**. The Add SIP ISUP profile dialog box appears.
- Name**—Enter the name of the SIP ISUP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required and has no default value.



5. Click **OK**. The new SIP ISUP profile appears in the SIP ISUP profile table.
6. Click the new SIP ISUP profile and click **Edit**. The Edit SIP ISUP profile window appears.
7. **ISUP version**—Click the ISUP version you want to convert in the drop down list.



8. **Convert outgoing messages to ISUP format**—Click the check box to enable SIP ISUP format version interworking. The default is disabled.



9. Click **Apply**.

Applying SIP ISUP Profiles

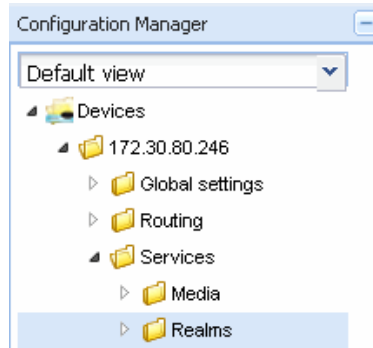
When you want to enable this feature for a realm, session agent, or SIP interface, you configure the **SIP ISUP profile** parameter with the name of the profile you want applied.

The following sections show you how to apply SIP ISUP profiles to a realm, session agent, or SIP interface.

Realm

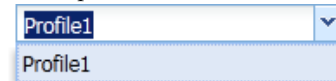
To apply a SIP ISUP profile to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the SIP realm you want to configure and click **Edit**. The Edit Realm window appears.
4. Click the Realm tab and scroll to SIP ISUP profile.
5. **SIP ISUP profile**—Click the name of the SIP ISUP profile, which defines the ISUP version to convert to. This is the value you entered in the **Name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP profile

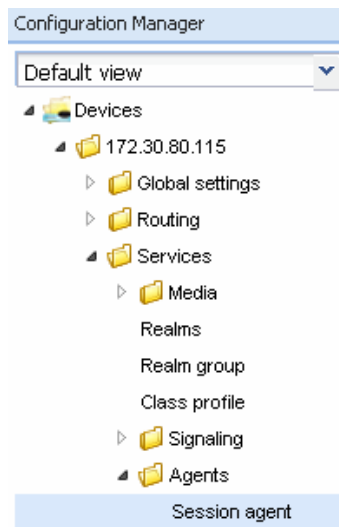


6. Click **Apply**.

Session Agent

To apply a SIP ISUP profile to a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent you want to configure and click **Edit**. The Edit session agent window appears.
4. Click the SIP session agent tab and scroll to SIP ISUP profile.
5. **SIP ISUP profile**—Click the name of the SIP ISUP profile, which defines the ISUP version to convert to. This is the value you entered in the **Name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

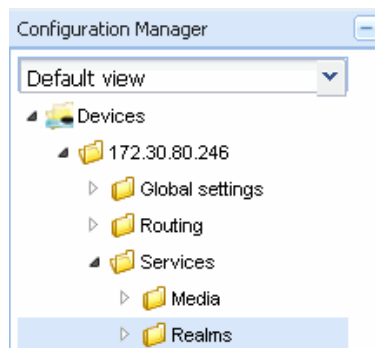


6. Click **Apply**.

SIP Interface

To apply a SIP ISUP profile to a SIP interface:

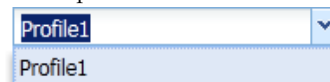
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the SIP realm you want to configure and click **Edit**. The Edit Realm window appears.

4. Click the Services/Protocols tab.
5. Click the SIP service/protocol you want to configure and click **Edit**. The Edit SIP service window appears.
6. Click the SIP interface tab and scroll to SIP ISUP profile.
7. **SIP ISUP profile**—Click the name of the SIP ISUP profile, which defines the ISUP version to convert to. This is the value you entered in the **Name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP profile


 A screenshot of a web interface showing a dropdown menu. The menu is open, displaying a list with 'Profile1' at the top. The text 'Profile1' is also visible in the input field above the dropdown arrow.

8. Click **Apply**.

IMS Charging Headers

The Net-Net SBC supports IMS Charging Headers. These headers include P-Charging-Vector and the P-Charging-Function-Address. IMS charging header support is configured separately from other IMS functions in order to support a variety of customer needs. Charging header information is now recorded in the CDR records.

A charging vector is defined as a collection of the charging information defined in RFC 3455. It is used to correlate charging records among network elements. The charging vector is constructed during the establishment of the dialog or a standalone transaction outside of a dialog.

Charging headers are inserted, deleted, or ignored for request messages. They are forwarded through the Net-Net SBC unmodified when embedded in response messages. If you wish to modify the charging headers in a response message, you must use the Net-Net SBC's header manipulation feature as a general solution.

P-Charging-Vector

You can configure the Net-Net SBC to process the P-Charging Vector header in three different ways.

- If a P-Charging-vector header is present in an incoming SIP request, the Net-Net SBC can pass the header untouched, as part of the full SIP message that is forwarded out of an egress interface.
- If a P-Charging-vector header is present in an incoming SIP request, the Net-Net SBC can delete the header and forward the full SIP message out of an egress interface.
- If an incoming SIP request does not contain a P-charging-vector header, the Net-Net SBC can create and insert the header and forward the full SIP message out of an egress interface. Likewise, if an incoming SIP request contains an existing P-Charging-Vector header, the Net-Net SBC can overwrite this header with the values generated internally.

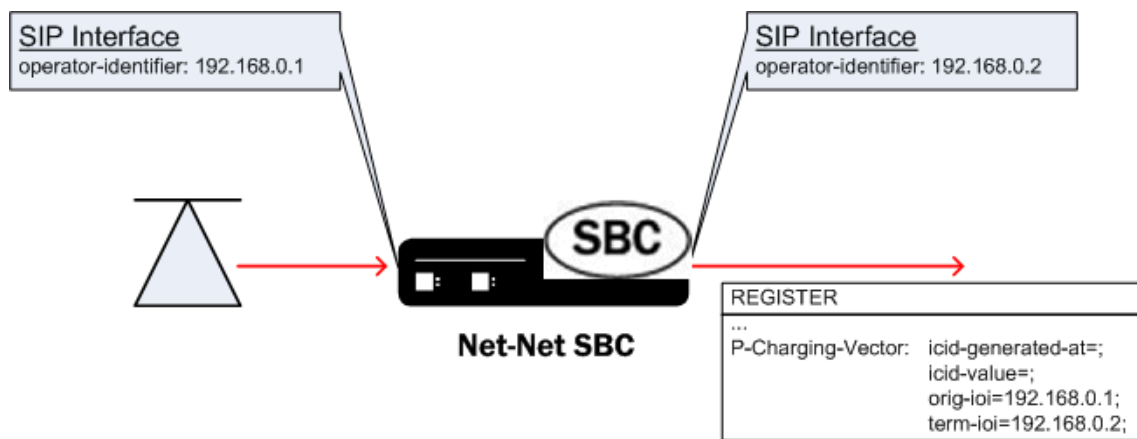
The P-Charging-Vector header is composed of four parameters: icid-value, icid-gen-addr, orig-ioi, term-ioi. See RFC 3455, Section 4.6 for more information.

1. The Net-Net SBC constructs the icid-value in the following format:
string2@string1 where:
String 1 is the IP address of the egress SIP interface.

String 2 is a unique string value created by the Net-Net SBC and based on the realm, local IP port, time, and a sequence number.

2. The `icid-gen-addr` parameter's value is the IP address of the egress SIP interface. This value is generated by the Net-Net SBC.
3. The `orig-ioi` parameter's value is set manually using the operator-identifier field located in the SIP interface configuration element.
4. The `term-ioi` parameter's value is set manually using the operator-identifier field located in the SIP interface configuration element.

You configure charging vector handling on the Net-Net SBC interface that receives the SIP request by turning on the switches that enable charging vector processing on the ingress interface for the call. Based on the direction of the call, the Net-Net SBC will insert the operator-identifier configuration parameter into the `orig-ioi` and the `term-ioi` parameters. The `orig-ioi` parameter takes the value of the operator-identifier configuration parameter of the SIP interface that receives the SIP request. The `term-ioi` parameter takes the value of the operator-identifier configuration parameter of the SIP interface that sends the SIP request to its next hop.



P-Charging-Vector Header Example

```

P-Charging-Vector: icid-
value=1ate6g46n1823s8719ck3ps6gbt46m5d3bci3po5hhdg3n86g1csio47g9c43@1
92.168.0.2;
icid-generated-at=192.168.0.2;
orig-ioi=192.168.0.1;
term-ioi=192.168.0.2;
  
```

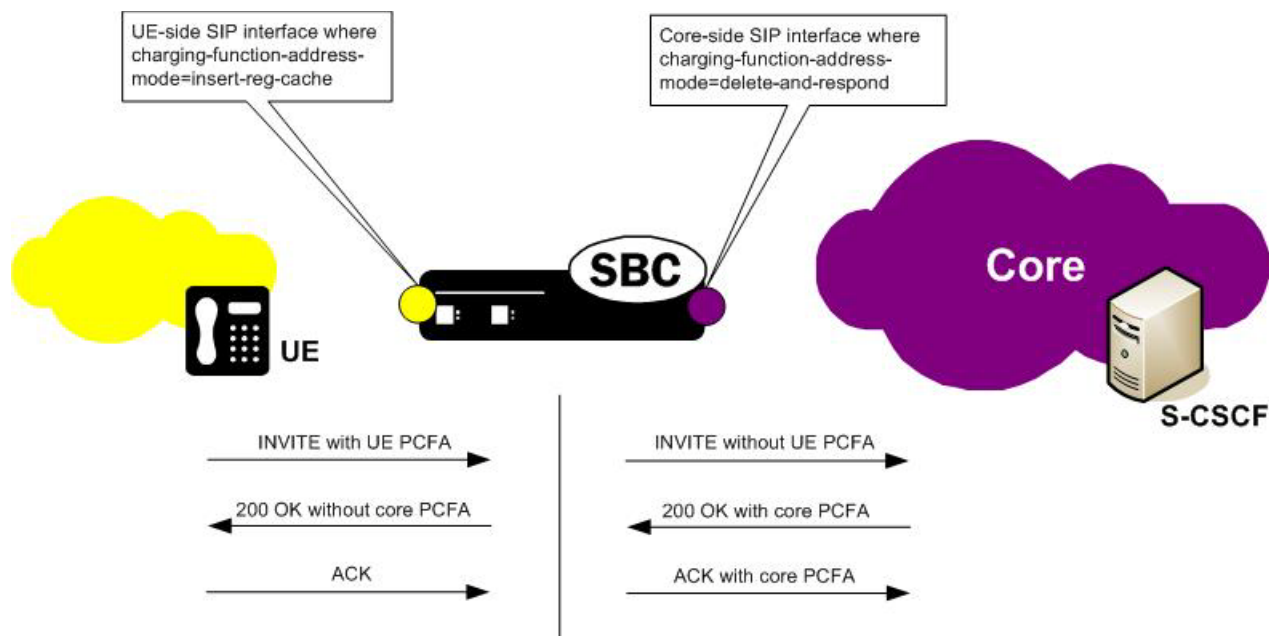
P-Charging-Function-Address

The P-Charging-Function-Address header is composed of two configurable parameters: `ccf`, `ecf`. You can configure the Net-Net SBC to process the P-Charging-Function-Address header in three different ways.

- If a P-Charging-Function-Address header is present in an incoming SIP request, the Net-Net SBC can be set to pass the header, untouched, as the full SIP request is forwarded out of an egress interface.
- If a P-Charging-Function-Address header is present in an incoming SIP request, the Net-Net SBC can be set to delete the header and forward the SIP request out of an egress interface.
- If an incoming SIP request does not contain a P-Charging-Function-Address header, the Net-Net SBC can be set to create and insert the header and forward the SIP message out of an egress interface.

- If an incoming SIP request contains a P-Charging-Function-Address header, and the Net-Net SBC is set to insert a configured P-Charging-Function-Address header, the new parameters will be appended before the existing parameters in the header. The Net-Net SBC will then forward the SIP request out of an egress interface.

In addition, the Net-Net SBC can also be configured to perform insertion and caching for the PCFA header in dialog-creating or stand-alone messages. The following diagram illustrates how this works:



For this scenario, there are two main functions, PCFA insertion and PCFA caching:

- **PCFA insertion**—Using the insert-reg-cache and delete-and-respond configuration values, the Net-Net SBC adds the PCFA to all SIP requests and to the response on the S-CPCF facing the SIP interface. However, only dialog-creating and standalone requests, and responses to each of those, update the Net-Net SBC and accounting information. Such requests do not have a To tag, and responses do not appear in established dialogs. The Net-Net SBC inserts the PCFA into provisional (1XX) and success (2XX) responses, with the exception of the 100 Trying response.

You can use SIP header manipulation rules (HMR) to remove any unwanted headers.

- **PCFA caching**—When you use either of the insert-reg-cache and delete-and-respond configuration values, the Net-Net SBC uses the latest cached copy of a PCFA header to insert into requests and responses. The Net-Net SBC does not cache any PCFA headers it receives on SIP interfaces using the none, insert, or insert-reg-cache modes because this type of SIP interface faces the UE making its replacement headers ones from the core.

Though there can be various sources for the latest cached copy, the PCFA header received as part of a dialog-creating or standalone request has highest precedence. This PCFA header is then stored as the latest cached value for that dialog. That is, for each specific dialog, the Net-Net SBC the PCFA is cached separately so it can add the most specific PCFA to the message—and is added to any message for the dialog.

When there is no cache PCFA for a specific dialog, the Net-Net SBC uses the registration cache entry as the latest cached copy. And when there is no entry in the registration, the PCFA uses the **ccf-address** and **ecf-address** values from the SIP interface.

The latest cached copy or the **ccf-address** is the value reported in the RADIUS VSA Acme-Session-Charging-Function-Address; this VSA is used for both of the new modes. Note that only the **ccf-address** is reported in RADIUS records; the **ecf-address** is not.

P-Charging-Function-Address Header Example

P-Charging-Function-Address: ccf=192.168.0.20 ; ecf=192.168.0.21;

RADIUS Accounting of Charging Headers

When the Net-Net SBC creates either the P-Charging-Vector header or the P-Charging-Function-Address header, it inserts an entry in the RADIUS record to record the charging header data.

For a P-Charging-Vector header, the icid-value is saved to the P-Charging-Vector attribute in the radius record. If the Net-Net SBC does not create a P-Charging-Vector header, but it receives a SIP message that already has the P-Charging-Vector header with an icid-value, the existing icid-value is written to the RADIUS record.

For a P-Charging-Function-Address header, the first CCF value is saved to the P-Charging-Function-Address attribute. When the Net-Net SBC creates the P-Charging-Function-Address, the CCF value it inserts into the header is saved to the radius record. If the Net-Net SBC does not create a P-Charging-Function-Address header, but it receives a SIP message that already has the P-Charging-Function-Address with a CCF value, the existing CCF value is written to the RADIUS record.

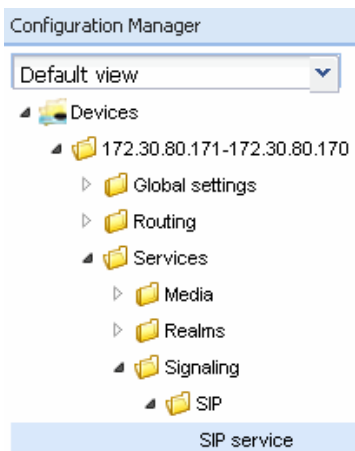
Name	Value	Value Type
Acme-Session-Charging-Vector	54	string
Acme-Session-Charging-Function-Address	55	string

Configuring P-Charging-Vector Processing for SIP Interfaces

P-Charging-Vector header handling is enabled in the SIP interface.

To configure P-Charging-Vector processing in a SIP interface:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP Interface you want to configure and click **Edit**. The SIP Interface window appears.
4. **Action to be performed on P-Charging-Vector header—Sets the Net-Net SBC to create the P-Charging-Vector header. The default value is pass. The valid values are:**
 - **none**—Pass this header unchanged (do not include icid-value in accounting records)
 - **pass**—Pass this header unchanged (include icid-value in accounting records)
 - **delete**—Delete this header
 - **insert**—Create this header
5. **Action to be perform on P-Charging-Function-Address header— Set the charging function address mode you want to use. The default value is pass. The valid values are:**
 - **none**—Pass the P-Charging-Vector header received in an incoming SIP message untouched as the message is forwarded out of the Net-Net SBC, but does not include icid-value in accounting records.
 - **pass**—Pass the P-Charging-Vector header received in an incoming SIP message untouched as the message is forwarded out of the Net-Net SBC, includes icid-value in accounting records.
 - **delete**—Delete the P-Charging-Vector header received in an incoming SIP message before it is forwarded out of the Net-Net SBC.
 - **insert**—Inserts the P-Charging-Vector header in an incoming SIP message that does not contain the P-Charging-Vector header. If the incoming message contains the P-Charging-Vector header, the Net-Net SBC will overwrite the P-Charging-Vector header with its values. This option always uses the **ccf-address** and **ecf-address** static values.
 - **insert-reg-cache**—To be configured on the SIP interface facing the UE, configures the Net-Net SBC to replace the PCFA with the most recently cached value rather than the **ccf-address** and **ecf-address** you set to be

static in your configuration. The cached values come from one of the following that the Net-Net SBC has received most recently: request, response, registration, or local configuration.

- **delete-and-respond**—To be configured on the SIP interface facing the S-CPCF, configures the Net-Net SBC to strip out the latest cached PCFA from the core side. The Net-Net SBC then remembers this PCFA and uses it in communications sent to the core.

Note that the default settings for this parameter and for **charging-vector-mode** are **pass** for new SIP interface configurations. If you are upgrading and there are pre-existing SIP interfaces in your configuration, the defaults become **none**.

6. **Charging collection function (CCF) address**—Set the CCF address value that will be inserted into the P-Charging-Function-Address header.
7. **Event charging function (ECF) address**—Set the ECF address value that will be inserted into the P-Charging-Function-Address header.
8. **Operator ID**— Set the operator identifier value to be inserted into a P-Charging-Vector header. The direction of the call determines whether this value is inserted into the orig-ioi or the term-ioi parameter in the P-Charging-Vector header. This string **MUST** begin with an alphabetical character.
9. Click **Apply**.

IMS: P-CSCF Endpoint Identification Using Address and Port

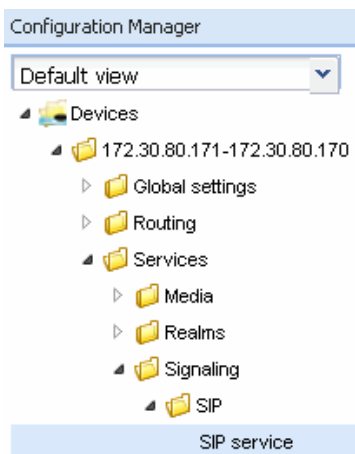
You can configure the Net-Net SBC, acting as a P-CSCF, to match a Request it receives to a registration cache entry based only on the IP address and port from which the Request came. When you enable this behavior, the Net-Net SBC will perform this kind of endpoint identification even when there nothing in the message matches the cache entry.

Configuring P-CSCF endpoint identification using address and port

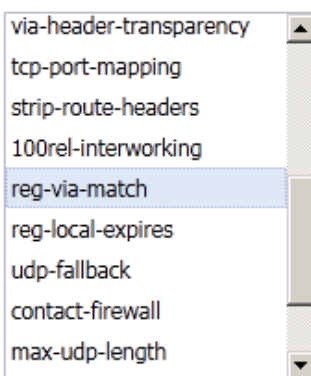
For this behavior to work as designed, you must also have the **reg-via-key** option enabled for the SIP interface to which you are adding the **reg-via-match** option.

To configure P-CSCF endpoint identification using address and port:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Scroll to the Options section and click **Options**.
4. **reg-via-match**— Choose the reg-via-select option from the Available options list.



5. Click **>>** to move the reg-via-match option to the Selected options list. The option name and description appear in the middle of the screen.
6. Click **OK**.
7. Click **Apply**.

1796 - SIP: Interim RADIUS Records for Recursive Attempts

When the Net-Net SBC routes calls, it performs local policy look-ups that can return several next hops, ordered by preference. This can also happen as a result of an LRT lookup, an ENUM query response, or SIP redirect. To set up sessions, the Net-Net SBC uses—in ordered preference—and recurses through the list if it encounters failures.

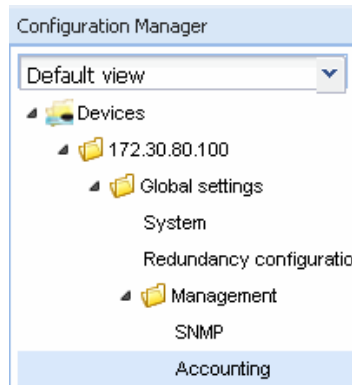
You can configure SIP accounting to send RADIUS Interim records when the Net-Net SBC encounters these failures. The interim message contains: the destination IP address, the disconnect reason, a timestamp for the failure, and the number that was called.

Please refer to the Net-Net 9000 Accounting Guide Release Version S-D7.1.0 to view the format of Intermediate local CSV CDRs in Appendix C.

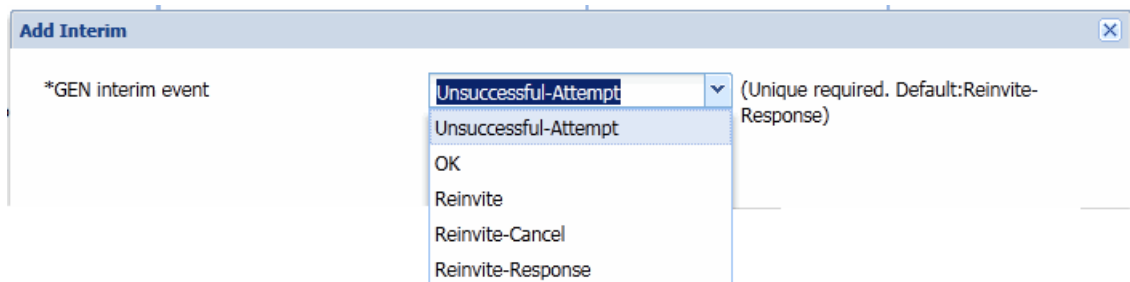
Interim RADIUS Records for Recursive Attempts

To configure SIP accounting to send Interim records for recursive attempts:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Scroll to Interim and click **Add**. The Add Interim window appears.
5. **GEN interim event**—Set this parameter to **unsuccessful-attempt** if you want the Net-Net SBC to send a RADIUS Interim records when a session set-up attempt from a preference-ordered list of next-hop destinations is unsuccessful. This can happen when a local policy lookup, LRT lookup, ENUM query response, or SIP redirect returns a preference-ordered list of next-hop destinations. The interim message contains: the destination IP address, the disconnect reason, a timestamp for the failure, and the number that was called.



6. Click **Apply**.

Expedited Media Termination of Calls with Abnormal Signaling Termination

When a remote endpoint does not receive a BYE message from the endpoint hanging up, and/or the Net-Net SBC does not receive a 200 OK message in response to the BYE from the remote endpoint, the remote call leg's media session is not closed gracefully. In such a case, the media session is closed by a media timer expiring.

In a normal call flow, the Net-Net SBC closes the remote call leg when it receives the 200 OK back from the remote endpoint. In the aforementioned non-standard case, the Net-Net SBC never receives this 200OK. If such situations are expected, you can configure the Net-Net SBC to close the remote call leg's media session as it forwards the BYE message to the remote endpoint, and not wait any further. This is accomplished by configuring the **release media at bye** option in the media manager.

Accounting Considerations

Even if the endpoint that has been hung up on does not respond immediately, the CDR created for the call must have the stop time reflect when the BYE was received by the Net-Net SBC, not when the response was received or when the SBC timed out the abandoned call leg.

To indicate cases like this, where the call termination cannot be completed by standard means, the Net-Net SBC includes the following combination of release/termination causes in the CDR:

Acct-Terminate-Cause = Session-Timeout

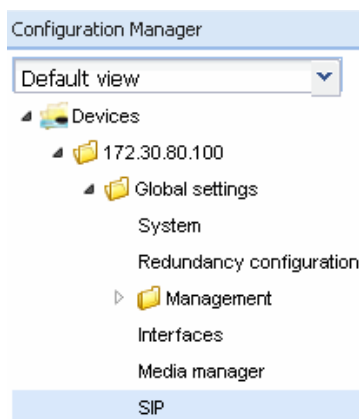
Acme-Disconnect-Cause = 47

Acme-SIP-Status = 908

Graceful Termination of Calls with Abnormal Signaling

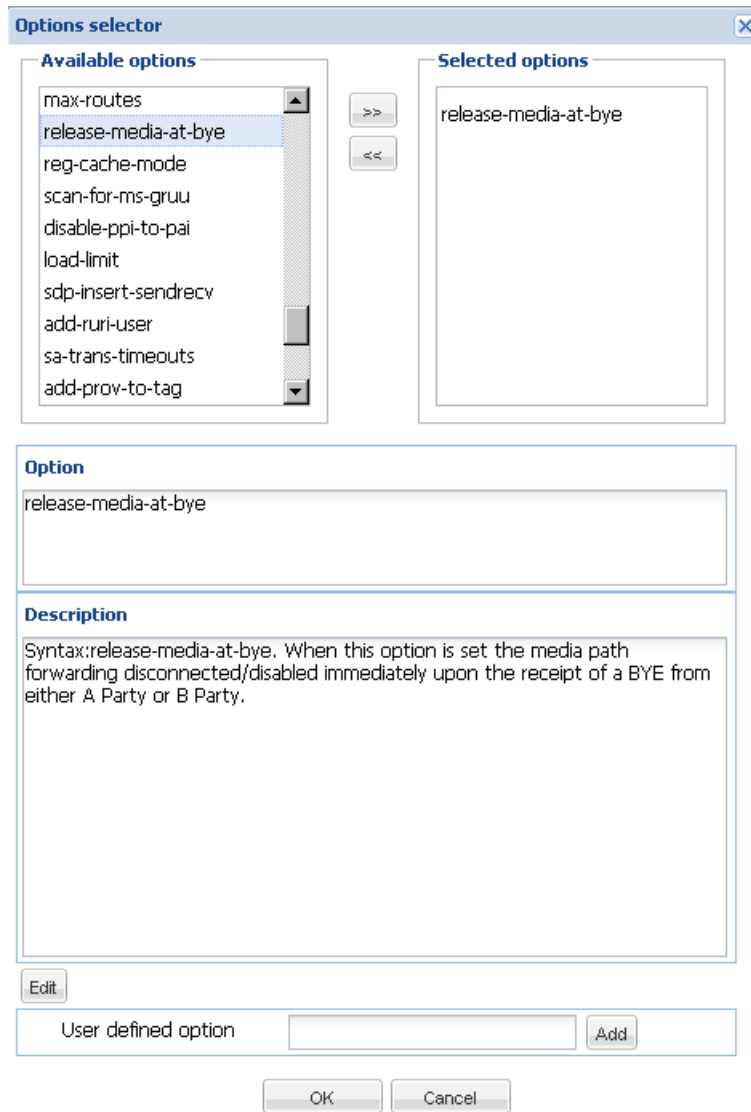
To configure graceful termination when signaling ended abnormally:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.

4. Click Options.
5. Select release-media-at-bye in the Available options list and click  to move the option to the Selected options list.



The image shows a screenshot of the 'Options selector' dialog box. It has a title bar with a close button. Inside, there are two main sections: 'Available options' and 'Selected options'. The 'Available options' section contains a list box with the following items: max-routes, release-media-at-bye (which is highlighted), reg-cache-mode, scan-for-ms-gruu, disable-ppi-to-pai, load-limit, sdp-insert-sendrecv, add-ruri-user, sa-trans-timeouts, and add-prov-to-tag. Between the two lists are two buttons: '>>' and '<<'. The 'Selected options' section contains a list box with the item 'release-media-at-bye'. Below these sections is a detailed view for the selected option. It has a label 'Option' with the text 'release-media-at-bye' in a text box. Below that is a 'Description' section with a text box containing the text: 'Syntax:release-media-at-bye. When this option is set the media path forwarding disconnected/disabled immediately upon the receipt of a BYE from either A Party or B Party.' At the bottom of the dialog, there is an 'Edit' button, a 'User defined option' text box with an 'Add' button, and 'OK' and 'Cancel' buttons at the very bottom.

Options selector

Available options

- max-routes
- release-media-at-bye
- reg-cache-mode
- scan-for-ms-gruu
- disable-ppi-to-pai
- load-limit
- sdp-insert-sendrecv
- add-ruri-user
- sa-trans-timeouts
- add-prov-to-tag

Selected options

- release-media-at-bye

Option

release-media-at-bye

Description

Syntax:release-media-at-bye. When this option is set the media path forwarding disconnected/disabled immediately upon the receipt of a BYE from either A Party or B Party.

Edit

User defined option Add

OK Cancel

The options detail appears in the description below.

6. Click OK.
7. Click **Apply**.

SIP Diversion to SIP-ISUP Interworking

For networks in which there are devices that do not support SIP-T or SIP-I (and support native SIP alone), the Net-SBC now supports SIP Diversion interworking. This feature enables such devices to function properly in instances that require SIP-T/SIP-I style ISUP IAM message encapsulation in ISUP requests, and to receive any call forwarding information in the IAM according to ISUP standards.

The Net-Net SBC interworks a native SIP INVITE request to SIP-T one by inserting an ISUP IAM body based on the INVITE; this includes redirections information based on the Diversion header. This feature can also perform the reverse translation. That is, it can interwork a SIP INVITE that does have the ISUP IAM body to a non-ISUP INVITE. In this case, the Net-Net SBC generates the necessary Diversion headers based on the IAM's Redirection information.

SIP Diversion to SIP-ISUP Interworking

To use this feature, you set up:

- sip-profile—Defines the redirection behavior
- sip-isup-profile—Defines the ISUP version to use when supporting SIP-T

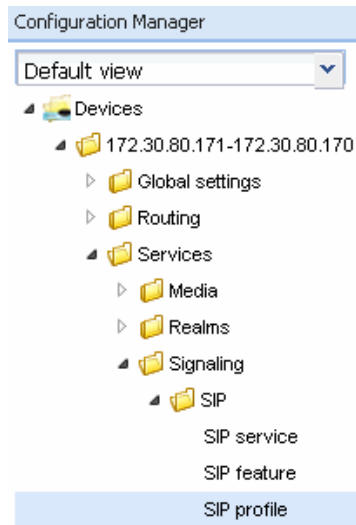
You can then apply these profiles to realms, session agents, and SIP interfaces where you want this feature enabled.

Configuring a SIP Profile

The **SIP Profile** configuration contains information that defines redirection behavior for the configuration where you apply it. You can set the redirection behavior to: **none**, **isup**, **inherit**, or **diversion**. You also uniquely identify the profile so that you can apply it by name in other configurations. This is a multiple-instance configuration, meaning that you can set up as many SIP profiles as needed.

To set up a SIP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.
3. Click Add. The Add SIP profile window appears.
4. **Name**—Enter the name of the SIP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.

*Name

5. Click **Apply**.
6. Select the SIP Profile you would like to use and click **Edit**.
7. **Redirection**—Choose the redirection mode you want to use: **inherit**(default), **isup**, **diversion**, or **none**. Note that when you set this parameter to **isup**, you should configure along with it a SIP ISUP profile; this will avoid any possible incompatibility when support for this feature expands (as expected).

Redirection

- isup
- diversion
- none
- inherit

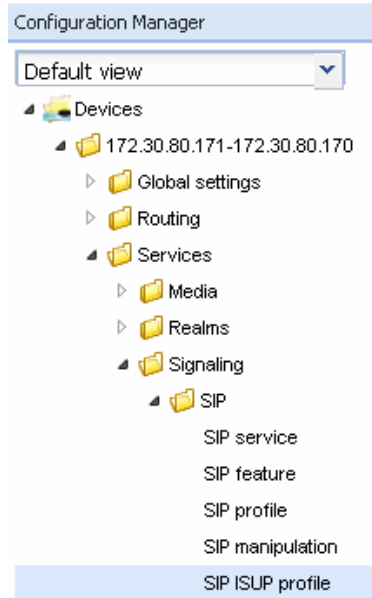
8. Click **Apply**.

Configuring a SIP ISUP Profile

The **SIP ISUP Profile** configuration contains information about what ISUP version you want to use. You also uniquely identify the profile so that you can apply it by name in other configurations. This is a multiple-instance configuration, meaning that you can set up as many SIP ISUP profiles as needed.

To set up a SIP ISUP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP ISUP profile. The SIP ISUP profile table appears in the content area.
3. Click **Add**.
4. **Name**—Enter the name of the SIP ISUP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.

*Name

5. Click **Apply**.
6. Select the SIP ISUP Profile you would like to use and click **Edit**.
7. **ISUP version**—Specify the ISUP version you want used in this profile in order to support SIP-T: **ansi-2000** (default) or **itu-99**.

ISUP version

8. Click **Apply**.

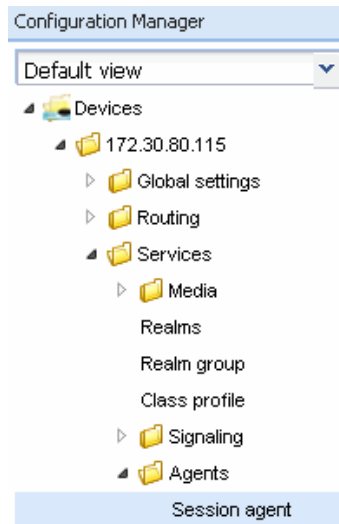
Applying SIP Profiles and SIP ISUP Profiles

When you want to enable this feature for a realm, session agent, or SIP interface, you configure the **SIP Profile** and **SIP ISUP Profile** parameters with the name of the profile you want applied.

The sample here shows this feature being applied to a session agent, but the realm and SIP interface configurations also have the same two parameters you use to set up the feature.

To apply a SIP profile and a SIP ISUP profile to a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. **SIP profile**—Select the name of the SIP profile, which defines the redirection behavior, from the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

Profile1	▼
Profile1	
Profile2	

4. **SIP ISUP profile**—Select the name of the SIP ISUP profile, which defines the ISUP version to use for SIP Diversion SIP-ISUP interworking, from the combination box. This is the value you entered in the **name** parameter of the SIP ISUP profile configuration. This parameter has no default value.

SIP ISUP profile

Profile1	▼
Profile1	

5. Click **Apply**.

SIP Registration Via Proxy

The Net-Net SBC supports a number of features that require it to cache registration information for UAs (endpoints) registering and receiving requests through it. For those features to operate correctly, the Net-Net SBC must act as the outbound proxy through which these endpoints register.

In order to support deployments where a proxy sits between the Net-Net SBC and the endpoints, the Net-Net SBC must consider the bottom Via header it receives from endpoints when constructing and matching cache registration entries. And when you use SIP port mapping, the system must use the bottom Via header as a way to determine the endpoint uniquely so that it can have a unique mapping port when the SIP interface is configured with the `reg-via-key=all` option.

Using the `reg-via-proxy` option, you can enable your Net-Net SBC to support endpoints that register using an intervening proxy. You can set this option either for a realm or for a SIP interface. If you use it for a SIP interface, add to the SIP interface pointing toward the proxy and endpoints—the access side.

Considerations for Reg-Via-Key and Port Mapping

When you set the `reg-via-proxy` option, the Net-Net SBC includes the bottom Via header from received requests in the registration cache Via Key. The system also uses it for determining whether or not the request matches a registration cache entry. Each unique bottom Via received a unique mapping port when you turn SIP port mapping on and set the SIP interface with the `reg-via-key=all` option.

Request Routing

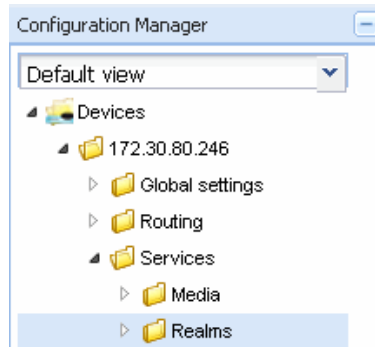
So that requests addressed to the corresponding registered contact are routed to the proxy, the Net-Net SBC includes the intervening proxy (i.e., the top Via) in the routing information for the registration cache when you set `reg-via-proxy`. To carry out this routing scheme, the system adds a Path header (if none is present) to the REGISTER. But it removes the Path header prior to sending the REGISTER to the registrar.

Note that when the received REGISTER contains a Path header, the Net-Net SBC uses it for routing requests toward the endpoint and includes it in the forwarded REGISTER request—as is the case when you do not enable SIP registration via proxy.

Configuring SIP Registration Via Proxy

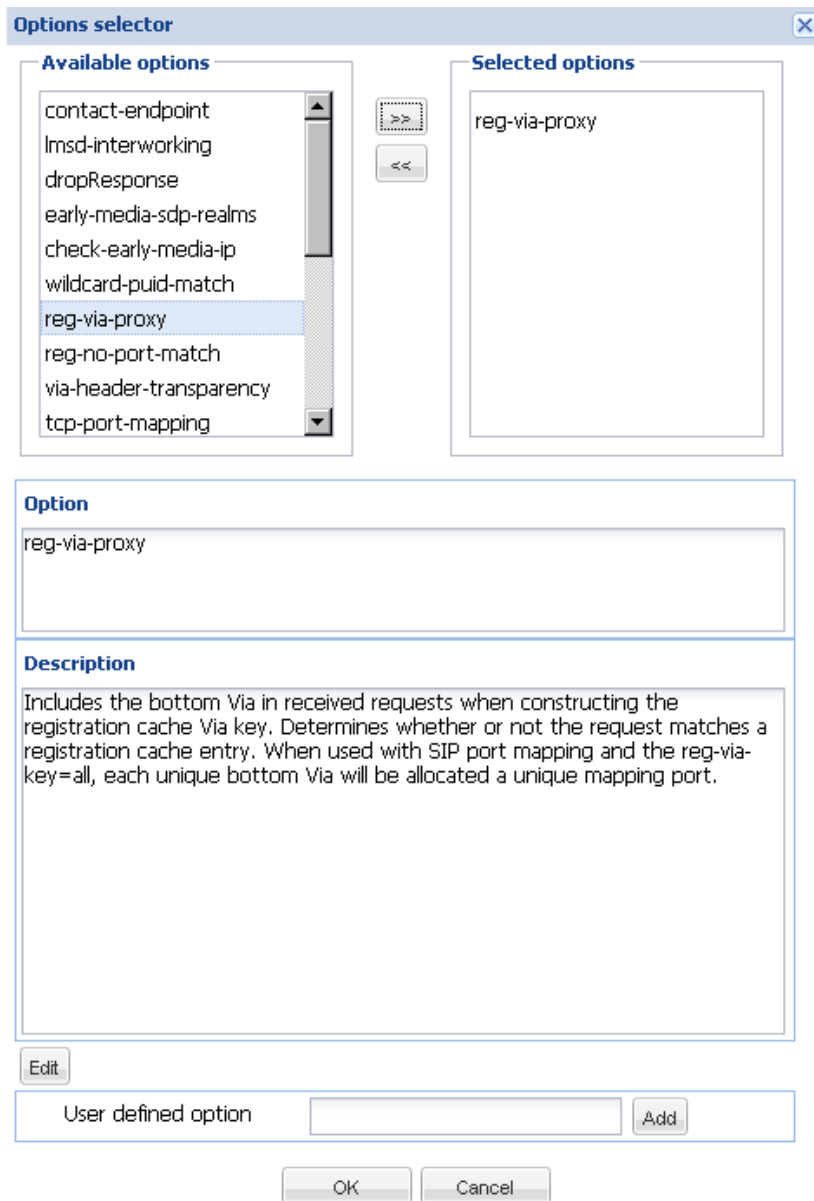
To configure SIP registration via proxy:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the realm table and click **Edit**.
4. Click the Services/Protocols tab.
5. Click the SIP service/protocol you want to configure in the services/protocols table and click **Edit**.
6. Click the General tab.
7. Scroll to Options.
8. Click **Options**. The Options window appears.
9. Click **reg-via-proxy** in the Available options list to select it.

10. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** section contains a list box with the following items: contact-endpoint, lmsd-interworking, dropResponse, early-media-sdp-realms, check-early-media-ip, wildcard-puid-match, **reg-via-proxy** (highlighted), reg-no-port-match, via-header-transparency, and tcp-port-mapping. Between the two list boxes are two buttons: a dashed border button with a right arrow and a solid border button with a left arrow. The **Selected options** section contains a list box with the item **reg-via-proxy**. Below these sections is a detailed view for the selected option. It has a label **Option** followed by a text field containing 'reg-via-proxy'. Below that is a label **Description** followed by a text area containing the text: 'Includes the bottom Via in received requests when constructing the registration cache Via key. Determines whether or not the request matches a registration cache entry. When used with SIP port mapping and the reg-via-key=all, each unique bottom Via will be allocated a unique mapping port.' Below the description is an **Edit** button. At the bottom of the dialog is a section for adding new options, with a label 'User defined option', a text field, and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

11. Click **OK**.
12. Click **Apply**.

SIP LMSD Interworking

A flavor of SIP, the Legacy Mobile Station Domain (LMSD) supports existing Mobile Stations (MSs) in a network that supports IP bearers. LMSD uses the Alert-Info header in 180 Ringing responses to INVITEs to tell User Agents that specific tones (or ringback) are to be generated locally by the UAC instead of passing on media received from the UAS. When the 180 Ringing response includes SDP, most UAs that do not support LMSD ignore the Alert-Info and do not play ringback locally to the calling party; they expect in-band media from the far-end. As a result, the caller hears silence instead of ringing until the call is answered.

The Net-Net SBC's support for LMSD enables it to suppress SDP in the 180 Ringing response so the UAC will play local ringback. This feature also covers cases where the Alert-Info from an LMSD UAS specifies the busy/congestions tone.

You enable this feature for individual SIP interfaces that face LMSD user agents by setting the **lmsd-interworking** option. For example, if endpoints supporting LMSD care in the core realm, then you add the **lmsd-interworking** option to the core SIP interface.

About Alert-Info Processing

The content from an Alert-Info header can be a URI (in the form `<http://LMSD/tone?sig-id=tone>`) to indicate tone or in the form `http://null` to indicate that media will be provided in-band from the far end. To account for variations in implementation, the Net-Net SBC looks only for **sig-id=** or for **/null** to determine the tone type. The following sig-id values yield the corresponding tones:

- **rt=ringtone**
- **bt=busy**
- **ct=congestion**

The Net-Net SBC treats all other sig-id values as ringtone.

When the Alert-Info contains neither **sig-id= no /null**, the system behaves as though there were no Alert-Info. In other words, it ignores the Alert-Info.

About LMSD Processing

The Net-Net SBC uses LMSD on the egress side for an INVITE client transaction where the **lmsd-interworking** option is present on the egress SIP interface. Since LMSD endpoints are required to use reliable provisional responses, 100rel-interworking is used on the when the received INVITE does not contain a Supported or Require header with a 100rel option tag. In this case, the system adds the 100rel tag to the Supported header in the INVITE it sends toward the LMSD endpoint. And the system removes the Rack header and the 100rel tag from the response it sends to the UAC.

180 Ringing

Once the Net-Net SBC determines if the Alert-Info specifies ringtone, busy/congestion, or nothing, it proceeds in the following ways for each case:

- **Ringtone**—When the Alert-Info indicates or is treated as ringtone, the Net-Net SBC flags the client transaction as having an LMSD ringtone. It saves the SDP—if any is present—in the response in the client-side dialog and removes it from the response it sends to the UAC. The saved SDP might be used for the 2xx response to the INVITE. Also, the client-side dialog is marked for early SDP

suppression to prevent the Net-Net SBC from forwarding an UPDATE request with a new SDP offer to the calling UA before a 2xx response arrives.

- **Busy/congestion**—When the Alert-Info indicates busy or congestion, the Net-Net SBC changes the 180 Ringing response to an error response. It uses 486 for busy and 503 for congestion. If the 180 Ringing is a reliable response, then the Net-Net SBC generates a PRACK to acknowledge it. The system will also send a CANCEL toward the UAS to cancel the INVITE transaction. The 486 or 503 error response is sent to the UAC after the removal of all message bodies and any Record-Route, Contact, Require: 100rel, or RSeq headers. The UAC should then generate the proper tone since it has not passed the UDP answer.
- **Null**—When there is no Alert-Info—or when the Net-Net SBC interprets there to be none—and the response contains SDP, LMSD is disabled for the remainder of the client transaction. And the Net-Net SBC will not check subsequent 180 responses for Alert-Info headers. The system will include an SDP answer in its response to the UAC, and the client-side dialog will no longer be marked for early SDP suppression.

Non-180 Responses

When it receives a on-180 provisional response with SDP, the Net-Net SBC disables LMSD for the remainder of the client transaction. No subsequent 180 responses will be checked for Alert-Info headers. Instead, the Net-Net SBC includes an SDP answer in the provisional response it sends to the UAC. In addition, the client-side dialog will no longer be marked for early SDP suppression.

UPDATE Request

If it sent an answer to the INVITE's SDP offer in a reliable provisional response, the UA can send a new SDP offer in an UPDATE request during an early dialog. In this case, "early" refers to a time before the INVITE 2xx response has been sent. If that SDP answer has been suppressed because of LMSD interworking, the Net-Net SBC must respond locally to the UPDATE in order to comply with SDP offer-answer rules. It cannot send the UPDATE to the calling UA.

Here, the Net-Net SBC checks to see if the dialog is marked for early SDP suppression. If it is, the Net-Net SBC generates a 200 OK response to the UPDATE and includes in it the SDP sent to the called UA in the original INVITE. The Net-Net SBC also saves the new SDP offer from the UPDATE in the client-side dialog, replacing any SDP saved previously. Then, if the SDP needs to be added to the INVITE 2xx response, it will be based on the latest SDP from the called UA.

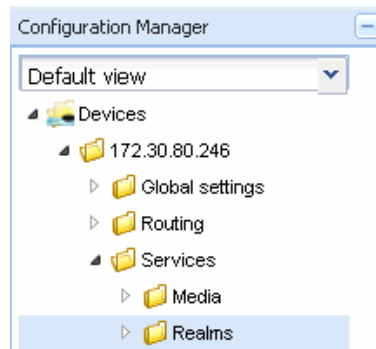
2xx INVITE Response

When a final success or 2xx response (usually a 200 OK) is received for the INVITE, it might not contain an SDP answer if the UAS had sent one previously in a reliable provisional response. If LMSD processing prevented an SDP answer from being sent to the UAC (in a reliable provisional response), an SDP answer must be added to the final 2xx response. If the SDP answer was saved in the client-side dialog, it will be added and the Net-Net SBC will update its flows accordingly. The resulting SDP answer will be part of the 2xx response destined for the UAC.

Configuring LMSD SIP Call Progress Tone Interworking

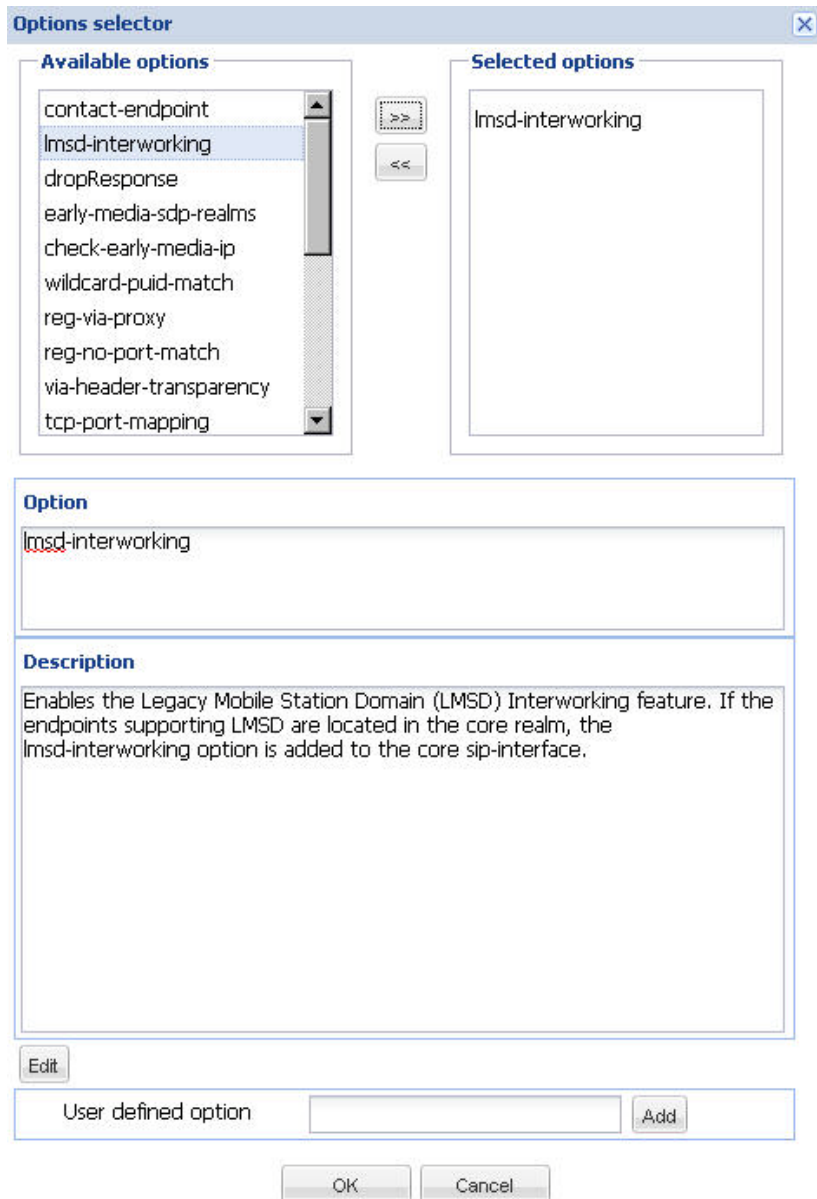
To enable SIP LMSD interworking for a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the realm table and click **Edit**.
4. Click the Services and Protocols tab.
5. Click the SIP service type you want to configure in the table and click **Edit**.
6. Click the General tab.
7. Scroll to Options.
8. Click **Options**. The Options window appears.
9. Click **lmsd-interworking** in the Available options list to select it.

10. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** section contains a list box with the following items: contact-endpoint, lmsd-interworking (highlighted), dropResponse, early-media-sdp-realms, check-early-media-ip, wildcard-puid-match, reg-via-proxy, reg-no-port-match, via-header-transparency, and tcp-port-mapping. Between the two sections are two buttons: a right arrow button (labeled with '>>') and a left arrow button (labeled with '<<'). The **Selected options** section contains a list box with the item 'lmsd-interworking'. Below these sections, there is a detailed view for the selected option. It has a label **Option** followed by a text field containing 'lmsd-interworking'. Below that is a label **Description** followed by a text area containing the text: 'Enables the Legacy Mobile Station Domain (LMSD) Interworking feature. If the endpoints supporting LMSD are located in the core realm, the lmsd-interworking option is added to the core sip-interface.' At the bottom of the dialog, there is an **Edit** button, a text field labeled 'User defined option', and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

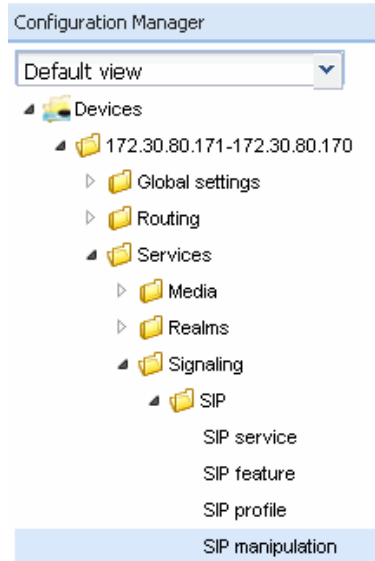
11. Click **OK**.
12. Click **Apply**.

HMR for SDP Offer-Answer

Configuring HMR SDP Offer-Answer

To configure HMR SDP Offer-Answer:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.
2. Click SIP manipulation.



3. Click the SIP manipulation name you want to configure and click **Edit**.
4. Scroll to Rule elements and click **Add**. The Select type to add dialog box appears.
5. **Service**—Click **mimeSdpRule** in the drop-down list.



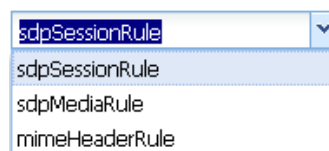
6. Click **Apply**. The Add rule elements dialog box appears.
7. **Name**—Enter the name for this rule element.
8. A Success dialog box appears.
9. Click **OK**. The new rule element is added to the Rule element table.

Configuring MIME SDP Rules

To configure a MIME SDP rule:

1. Click the rule element you want to configure and click **Edit**.
2. Scroll to the Session rules, Media rules and MIME header rules table and click **Add**. The Select type to add dialog box appears.
3. **Service**—Click sdpSessionRule in the drop-down list.

Service



4. Click **Apply**. The Add Session rules, Media rules and MIME header rules dialog box appears.
5. **Name**—Enter a unique name for this instance.
6. Click **Apply**. A Success dialog box appears.
7. Click **OK**.

Configuring Line Rules

To configure a line rule:

1. Click the session rule, media rule and MIME header rule instance you want to configure in the table and click **Edit**.
2. Click **Add** in the Line rules table. The Add line rules dialog box appears.
3. **Name**—Enter a unique name for this line rule.

*Name

line-rule01

4. Click **Apply**. A Success dialog box appears.
5. Click **OK**.

Wildcarded PUID in P-Associated-URI

In the IMS core, a PBX often registers its own base telephone number with the S-CSCF, and the S-CSCF obtains a wildcarded public user identity, represented by a regular expression that describes all extension numbers associated with the registering PBX.

The S-CSCF includes the wildcarded Public User Identifier (PUI) in the 200OK response to the REGISTER message that the Net-Net SBC sent it, prompted by the initial registering endpoint. Upon receiving the 200OK, the Net-Net SBC uses the wildcarded puid to implicitly treat subsequent INVITEs from PBXs extensions that match the wildcard as registered endpoints.

For all calls directed at telephone numbers that are implicitly registered by the wildcarded puid regular expression, the Net-Net SBC terminates them the servicing PBX.

This behavior is configured by adding the **wildcard-puid-match** option on the access side SIP interface when requests originate there.

Wildcarded Public User Identity

A wildcarded PUI consists of a delimited regular expression located either in the userinfo portion of the SIP URI or in the telephone-subscriber portion of the Tel URI. The regular expression in the wildcarded PSI shall take the form of Extended Regular Expressions (ERE) as defined in chapter 9 in IEEE 1003.1-2004 Part 1. The delimiter is the exclamation mark character ("!").

Using the following PUI as an example:

```
sip:chatlist!.*!@example.com
```

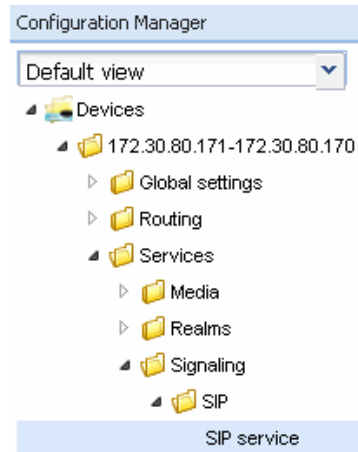
The following PUIs will match:

```
sip:chatlist1@example.com
sip:chatlist2@example.com
sip:chatlist42@example.com
sip:chatlistAbC@example.com
sip:chatlist!1@example.com
```

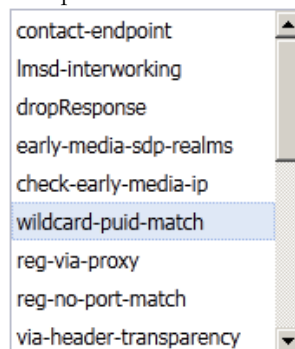
Wildcarded PUID Registration

To configure wildcarded PUID registration:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP Interface you want to configure and click **Edit**. The SIP Interface window appears.
4. Scroll to the Options section of the content area.
5. Click **Options**. The Options window appears.
6. Click wildcard-puid-match in the Available options list and click  to move the option to the Selected options list.



7. Click **OK**.
8. Click **Apply**.

Intrusion Detection System

Release S-C6.2.1 provides an enhanced Intrusion Detection System referred to within this document as IDS Phase 2, which supplements the original IDS reporting and protection services first offered in the Net-Net S-C6.2.0 release. IDS Phase 2 provides enterprise users with additional tools to identify, monitor, and control suspicious, and possibly, malicious traffic patterns.

License Requirements

IDS Phase 2 requires the IDS Advanced Reporting license, which is different from the IDS license introduced in the Net-Net S-C6.2.0 release. The IDS Advanced Reporting license enables both the initial IDS services offered by S-C6.2.0 and the Phase 2 enhancements offered by S-C6.2.1. Access to new services described in this Technical Notice requires that the original IDS license be upgraded to the IDS Advanced Reporting license.

Rejected SIP Calls

IDS Phase 2 provides tools to monitor and record rejected SIP calls. A sudden or gradual increase in such calls can, but need not, indicate malicious intent.

IDS Phase 2 provides a global counter that increments with each SIP INVITE or REGISTER message that is rejected by the Acme Packet Net-Net SBC, and offers the option of generating a syslog message in response to call rejection.

Rejected Calls Counter

The rejected calls counter is a 32-bit global counter that records the total number of rejected SIP calls. For the initial (S-C6.2.1 release) such calls have been rejected by the SBC with the following response codes: 400, 403, 404, 405, 408, 413, 416, 417, 420, 423, 480, 481, 483, 484, 485, 488, 494, 500, 503, 505, and 604. These response codes may change with future software revisions.

The current value of the rejected calls counter is accessible via SNMP, Historical Data Recording (HDR), or the ACLI.

apSysMgmtGeneralObjects Table (1.3.6.1.4.1.9148.3.2.1.1)

Object Name	Object OID	Description
apSysSipTotalCallsRejected	1.3.6.1.4.1.9148.3.2.1.1.25	Global counter for SIP calls that are rejected by the SBC

The *sip-error* HDR collection group contains a new reporting field, *Call Rejects*, which contains the value of the global rejected calls counter.

Syslog Reporting of Rejected Calls

Users can choose to send a syslog message in response to the rejection of a SIP call. In the default state, rejected calls are not reported to syslog.

The Net-Net EMS label: **Allow syslog to be sent for call that gets rejected**, which is disabled by default, enables the generation of a syslog message in response to the rejection of a SIP call.

Syslog messages issued in response to call rejection contain the following call-related information.

- SIP status code indicating rejection cause
- SIP method name (INVITE or REGISTER)
- Reason for denial
- Realm of calling endpoint
- Applicable local response map
- Content of Reason header (if present)
- From URI of calling endpoint
- Target URI of called endpoint
- Source and Destination IP address and port
- Transport type

The following are sample syslog messages issued in response to call rejections.

```
Dec 8 06:05:42 172.30.70.119 deimos sipd[205bfee4] ERROR [IDS_LOG]INVITE from
source 172.16.18.100:5060 to dest 172.16.101.13:5060[UDP] realm=net172; From=sipp
<sip:sipp@172.16.18.100:5060>;tag=13890SIPpTag001;
target=sip:service@172.16.101.13:5060 rejected!; status=483 (Too Many Hops)
```

```
Dec 10 15:09:28 172.30.70.119 deimos sipd[2065ace8] ERROR [IDS_LOG]INVITE from
source 172.16.18.5:5060 to dest 172.16.101.13:5060[UDP] realm=net172; From=sipp
<sip:sipp@172.16.18.5:5060>;tag=10015SIPpTag001;
target=sip:service@172.16.101.13:5060 rejected!; status=488 (sdp-address-mismatch);
error=sdp address mismatch
```

IDS syslog messages that report rejected calls (introduced as part of IDS Phase 2 in S-C6.2.1) and those that report endpoint demotions (introduced as part of IDS Phase 1 in S-C6.2.0) now contain a string “IDS_LOG”, to facilitate their identification as IDS-related messages. With IDS Phase 2, IDS messages reporting either endpoint demotions or call rejections can be sent to specific, previously-configured syslog servers.

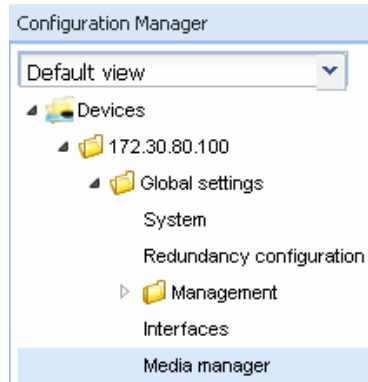
In topologies that include multiple syslog servers, use the following procedure to enable delivery of IDS-related messages to one or more specific syslog servers.

Enabling Syslog Reporting of Rejected Calls

Users can choose to send a syslog message in response to the rejection of a SIP call. In the default state, rejected calls are not reported to syslog.

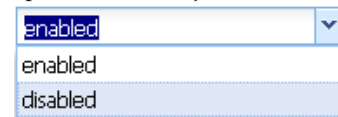
To enable syslog reporting of rejected SIP calls:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to **Syslog on call reject**.
5. **Syslog on call reject**—Click **enabled** in the drop-down list to enable generation of a syslog message in response to the rejection of a SIP call.

Syslog on call reject

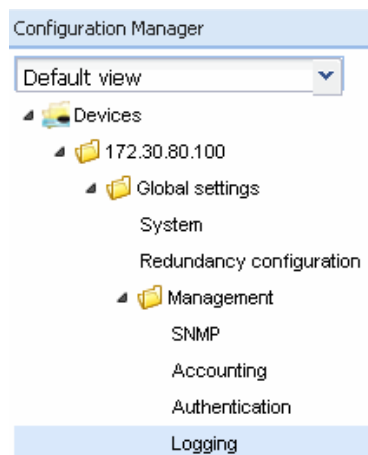


6. Click **Apply**.

Enabling Message Transfer to Specific Syslog Servers

To enable message transfer to specific syslog servers:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Logging. The Logging data appears in the content area.

3. Scroll to Syslog.
4. **IDS syslog facility**—Enter a value to enable message transfer to specific syslog servers. The default value, *-1*, disables selective message transfer. To enable transfer to a designated syslog server or servers, enter a value ranging from 0 through 23.

IDS syslog facility

-1

5. Click **Apply**.

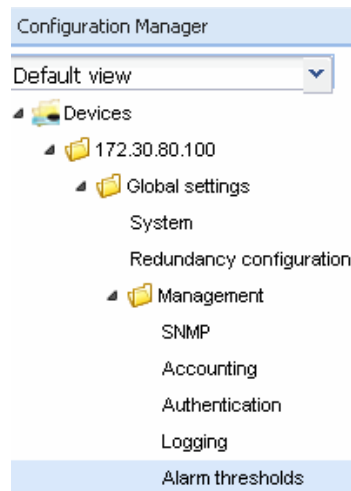
TCA Reporting of Denied Entries

You can configure a Threshold Crossing Alarm (TCA), which issues minor, major, and critical system alarms when the count of denied entries exceeds pre-configured values. For each issued alarm, the TCA also transmits an SNMP trap that reports the alarm state to remote SNMP agents.

Configuring Threshold Crossing Alarms

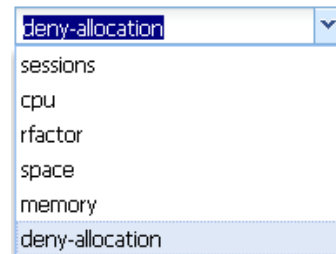
To configure a threshold crossing alarm:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Alarm thresholds. The Alarm threshold table appears in the content area.
3. Click **Add**. The Add alarm-threshold dialog box appears.
4. **Type**—Click **deny-allocation** in the drop-down list to specify the maximum number of denied entries allocated by the Net-Net SBC.

*Type



Threshold values that trigger alarm and SNMP trap generation are defined as percentages of the **deny-allocation** attribute.

5. Click **Apply**. A Success dialog box appears.

6. Click **OK**.
7. Click **Apply**.

After issuing a system alarm and accompanying SNMP trap, the TCA continues to monitor the number of denied entries. If the number of denied entries rises to the next threshold value, a new, and more severe, system alarm/SNMP trap is generated. If the number of denied entries falls below the current threshold level, and remains there for a period of at least 10 seconds, a new, and less severe system alarm/SNMP trap is generated.

Syslog Reporting of Denied Entries

Syslog reporting of endpoint demotions was introduced as part of IDS Phase 1 in S-C6.2.0. With IDS Phase 2 (S-C6.2.1 and later), such syslog messages contain the last SIP message from the endpoint that caused the transition to the denied state. If the included SIP message increases the length of the syslog beyond 1024 bytes, the SIP message is truncated so that the syslog is no larger than 1024 bytes.

CPU Load Limiting

The transmission of IDS-related system alarms and SNMP traps is disabled when the CPU utilization rate surpasses a configured threshold percentage. When the threshold is exceeded, a syslog message (MINOR level) announces the termination of IDS reporting. No additional syslog messages or SNMP traps are generated until the CPU utilization rate falls below the configured threshold. The resumption of IDS reporting is announced by another syslog message, also issued at the MINOR level.

By default the CPU utilization rate is 90%.

Denied Endpoints

IDS Phase 2 provides a denied endpoint counter that includes SIP and MGCP (Media Gateway Control Protocol) endpoints. The global counter value is available via SNMP or HDR.

The global counter value is available to SNMP under APSYSMGMT-MIB -> acmepacketMgmt -> apSystemManagementModule -> apSysMgmtMIBObjects -> apSysMgmtMIBGeneralObjects.

apSysMgmtGeneralObjects Table (1.3.6.1.4.1.9148.3.2.1.1)		
Object Name	Object OID	Description
apSysCurrentEndptsDenied	1.3.6.1.4.1.9148.3.2.1.1.26	Global counter for current endpoints denied

The *system* HDR collection group contains a new reporting field, *Current Deny Entries Allocated*, which contains the value of the global endpoints denied counter.

SIP Proxy Subscriptions

2024

When the Net-Net SBC operates in dialog mode (i.e., as a B2BUA), it creates and maintains dialog state for subscription dialogs created with SUBSCRIBE/NOTIFY messages and for INVITE-initiated dialogs. Since there can be a very large number of subscriptions per user in a Rich Communication Services (RCS) environment

(especially for “presence” subscriptions), Net-Net SBC resources can quickly become depleted.

To alleviate this consumption of resources, you can configure your Net-Net SBC to operate in proxy mode for event packages that you define using the **Proxy sub events** parameter in the global SIP configuration. When you define event packages in this list and the operation mode for the SIP configuration is dialog or session, the Net-Net SBC processes all SUBSCRIBE and NOTIFY requests and responses for the designated event packages in transaction stateful mode.

Topology Hiding

So that it can perform topology hiding, the Net-Net SBC retains necessary routing information (such as the Contact or Record-Route header values) and it encodes certain data from the messages it receives in the messages it sends. To be more specific, the Net-Net SBC encodes the original URI hostport and the ingress realm name into a “gr” parameter it adds to the URI. The hostport information is replaced with the IP address and port of the SIP interface from which the message is sent (the egress interface). Without this information, subsequent in-dialog messages cannot be routed correctly because the Net-Net SBC does not retain dialog state (i.e., the route-set or remote-target).

For example, the URI `sip:td@192.168.24.121:5060` might be encoded as `sip:td@192.168.24.121:5060; gr=vjml9qtd175bhmhvhkgp0jov81popvbp000040`.

The Net-Net SBC also performs URI encoding on the message body for Content-Type “application/pidf+xml”. This contains a Presence Information Data Format document (or PIDF) in PUBLISH and NOTIFY requests so subsequent SIP requests using the URIs in the PIDF document can be routed correctly.

The Net-Net SBC performs URI encoding in outgoing SIP messages after SIP=NAT is applied and before outbound HMR occurs. And the system decodes URIs in SIP messages after inbound HMR takes place and before SIP-NAT is applied.

In the event a URI is encoded several times (as is the case in spiral and hairpin calls), the encoded realm+hostport values are separated by a plus sign (+), as in the following:

`sip:td@192.168.24.121:5060; gr=vjml9qtd175bhmhvhkgp+dhfhhb0jov81opvbp`

How It Works

When the Net-Net SBC receives any of the following requests, it matches the contents of the request’s Event header with the list you configure in the proxy-sub-events parameter:

- PUBLISH
- SUBSCRIBE
- NOTIFY
- REFER

This is provided the operation-mode for the SIP configuration is set to either **session** or **dialog**. If it finds a match, the Net-Net SBC marks the request for processing in transaction-stateful mode rather than in B2BUA mode.

Note: Although PUBLISH is not a dialog-creating request, topology hiding needs to be applied to the PIDF so that subsequent NOTIFY requests containing portions of the published PIDF can be decoded properly.

When the Net-Net SBC forwards the request, it will have encoded all Contact and Record-Route header information using the ingress realm. The hostport value of the URIs then has egress SIP interface's IP address and port. The Via headers in the requested the Net-Net SBC received are not included in the outgoing request.

Then PUBLISH, SUBSCRIBE, NOTIFY, and REFER responses are compared to the request that was sent to determine if the response should receive transaction-stateful proxy treatment. The Net-Net SBC decodes any encoded Record-Route headers back to their original values for the outgoing response. Any Record-Route headers added downstream from the Net-Net SBC are encoded using the original request's egress realm (meaning the realm from which the response was received). In addition, the Contact header is encoded using the request's egress realm and ingress SIP interface and port.

Feature Interaction

When using this feature, the Net-Net SBC does not keep dialog or subscription state. Therefore the Per-user SUBSCRIBE Dialog Limit feature—configured in the **enforcement-profile** configuration—will not function properly when a subscription is handled in proxy mode.

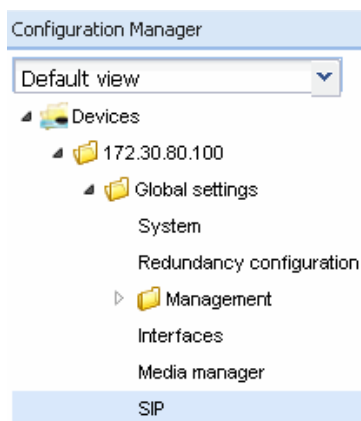
Configuring Proxy Sub events

This section shows you how to configure a list of SIP event packages to cause the Net-Net SBC to act in proxy mode.

Note: The **Operation mode** parameter for the global SIP configuration must be set to either **dialog** or **session in order** for this feature to function as designed.

To configure SIP event packages to enable SIP proxy subscriptions:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click the System folder.
5. Scroll to Proxy sub events.
6. Click **Add**. The Add proxy sub events dialog box appears.

7. ***Proxy sub-event**— Enter the SIP event package name that you want to enable the SIP proxy subscriptions feature.

8. Click **Apply**. A Success dialog box appears.
9. Click **OK**. The proxy sub event is added to the Proxy sub event table.
10. Repeat steps 5 through 8 to add multiple proxy sub events.
11. Click **Apply**.

SIP SDP Insertion for Re-INVITES

If your network contains some SIP endpoints that do not send SDP in ReINVITES but also contains others that refuse INVITES without SDP, this feature can facilitate communication between the two types. The Net-Net SBC can insert SDP into outgoing INVITE messages when the corresponding, incoming INVITE does not contain SDP.

You can also use this feature when the network devices used in H.323-SIP interworking do not include SDP in the INVITES sent to SIP endpoints. In this case, the Net-Net SBC can insert SDP in the outgoing INVITE messages it forwards to the next hop.

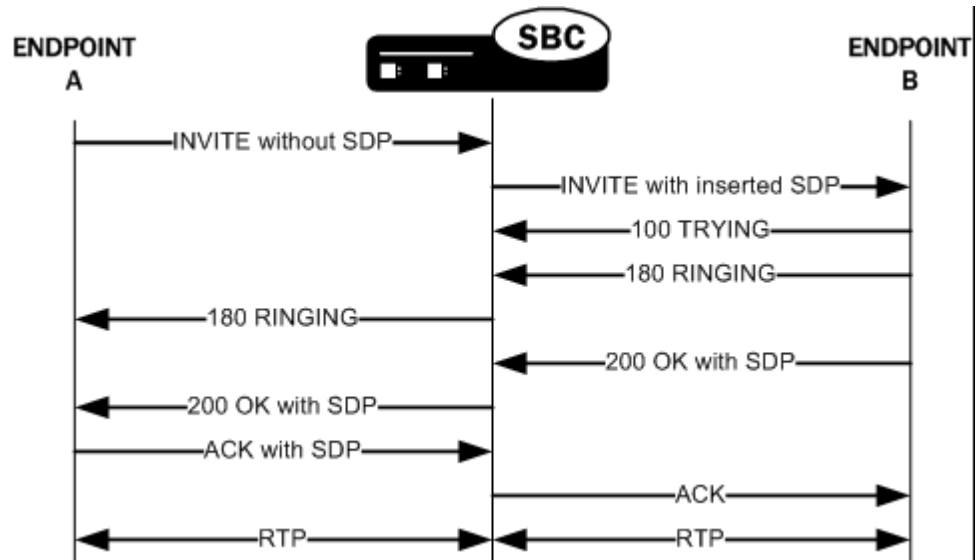
This feature works for both INVITES and ReINVITES.

How It Works

This section explains how the SDP insertion feature works for INVITES and ReINVITES. The examples used this section are both pure SIP calls. Even when you want to use this feature for IWF calls, though, you configure it for the SIP side.

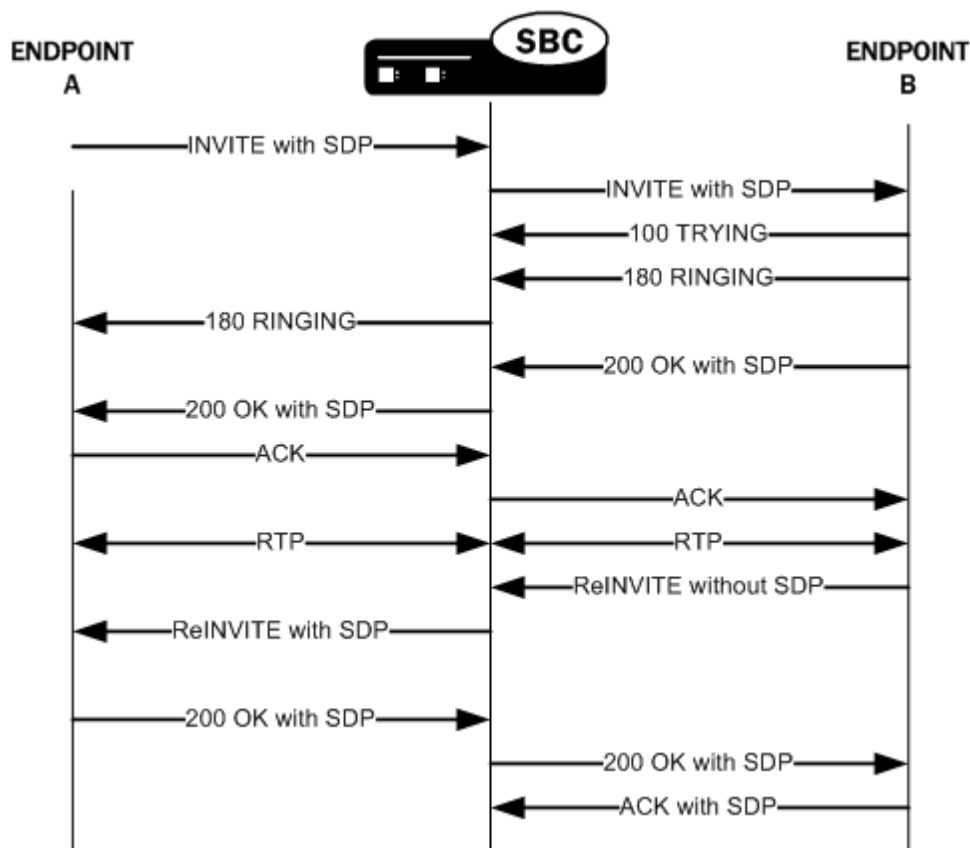
SDP Insertion for SIP INVITES

With the parameters mentioned above appropriately configured, the Net-Net SBC inserts SDP into an outgoing INVITE when the corresponding incoming INVITE has none. Because no SDP information is available for the session, the Net-Net SBC uses a media profile from a list of them you configure and then apply for SDP insertion.



SDP Insertion for SIP ReINVITES

The section explains SDP insertion for ReINVITES, using a case where SIP session has been established with an initial INVITE containing SDP. In the diagram below, you can see the initial INVITE results in a negotiated media stream. But after the media stream is established, Endpoint B sends a ReINVITE without SDP to the Net-Net SBC. In this case, the Net-Net SBC uses the negotiated media information from the initial INVITE to insert when the ReINVITE has no SDP. It then sends this ReINVITE with inserted SDP to the next hop signaling entity.

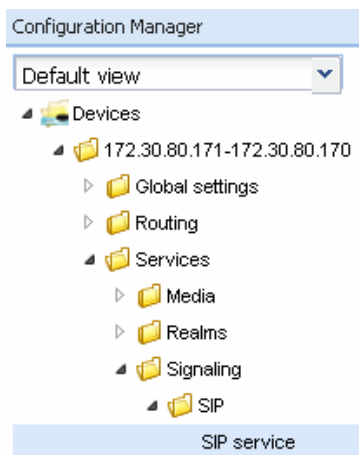


Configuring SDP Insertion for SIP INVITES

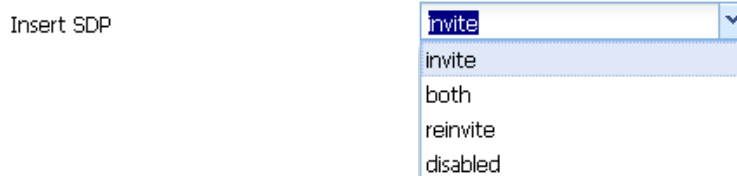
To work properly, SDP insertion for SIP invites requires you to set a valid media profile configuration. For more information about how to set up media profiles, see the *Net-Net Configuration Guide*.

To enable SDP insertion for INVITES:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



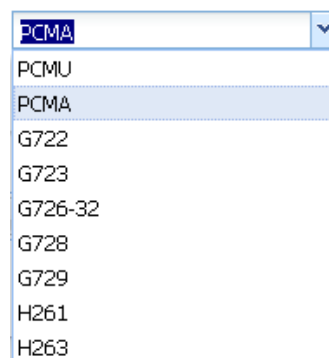
2. Click SIP service.
3. Click the SIP Interface tab.
4. Select the SIP interface you would like to configure and click **Edit**.
5. Click the General tab.
6. Scroll to SDP Insertion re-INVITE without SDP.
7. **Insert SDP**—Click **invite** in the drop down list.



8. Scroll to List of media profiles and click **Add**. The Add List of media profiles dialog box appears.
9. **Media profile**—Select the media profile that you want to use in the drop down list when the Net-Net SC inserts SDP into incoming INVITEs that have no SDP. The media profile contains media information the Net-Net SBC inserts in outgoing INVITE.

This parameter is empty by default.

*Media profile



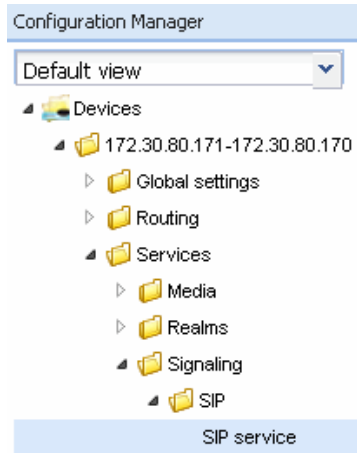
10. Click **Apply** to apply the media profile.
11. Click **Apply**.

Configuring SDP Insertion for SIP ReINVITEs

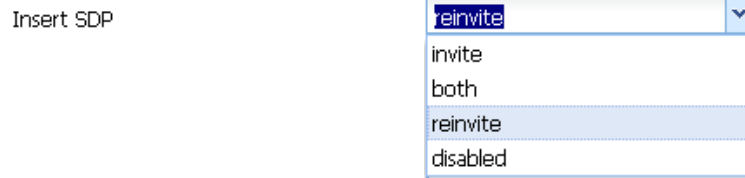
In this scenario, the Net-Net SBC uses the media information negotiated early in the session to insert after it receives an incoming ReINVITE without SDP. The Net-Net SBC then sends the ReINVITE with inserted SDP to the next hop signaling entity. You do not need the media profiles setting for ReINVITEs.

To enable SDP insertion for ReINVITEs:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Select the SIP interface you would like to use and click **Edit**.
4. **Insert SDP**—Click **reinvite** in the drop down list.



5. Click **Apply**.

SIP REFER with Replaces

To support enterprise and call center applications, the Net-Net SBC provides the ability for one party participating in a three-way call to request direct connectivity between the other two parties and to leave the call silently when that connectivity is established. SIP supports this function using the Replaces header in a REFER message, also known as “REFER with Replaces.”

The most common application of REFER with Replaces handling occurs in a high-level sequence like this:

1. The customer calls a customer service line and reaches—via the Net-Net SBC—an IVR/ACD (Interactive Voice Response system/Automatic Call Distribution system). In some architectures, these are two separate elements.
2. Based on the customer’s selection from the menu of options, the IVR/ACD contacts an agent via the Net-Net SBC.
3. Since the ultimate goal is for the IVR/ACD to drop out of the path, it sends a REFER with Replaces to the Net-Net SBC. This message indicates the Net-Net SBC should replace the IVR/ACD endpoint in the call leg with the agent’s endpoint.
4. The Net-Net SBC processes the REFER with Replaces, issuing ReINVITEs to the customer with the agent’s parameters.
5. The IVR/ACD drops out of the media path once the bridged call between the customer and the agent is established.

Note that direct media connectivity between endpoints must be possible in order for the REFER with Replaces to be carried out properly. For example, if both endpoints (such as the customer and agent from the example above) are behind the same firewall, direct media connectivity should be possible. However, if one endpoint is behind a firewall and the other is not, then direct media connectivity may not be possible.

For licensing capacity purposes, note that a bridged session counts as a single call.

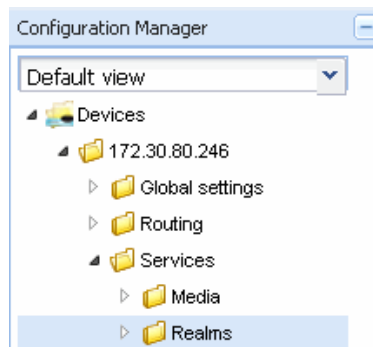
Configuring SIP REFER with Replaces

You enable SIP REFER with Replaces handling either in the realm configuration or in the session agent configuration.

Realms

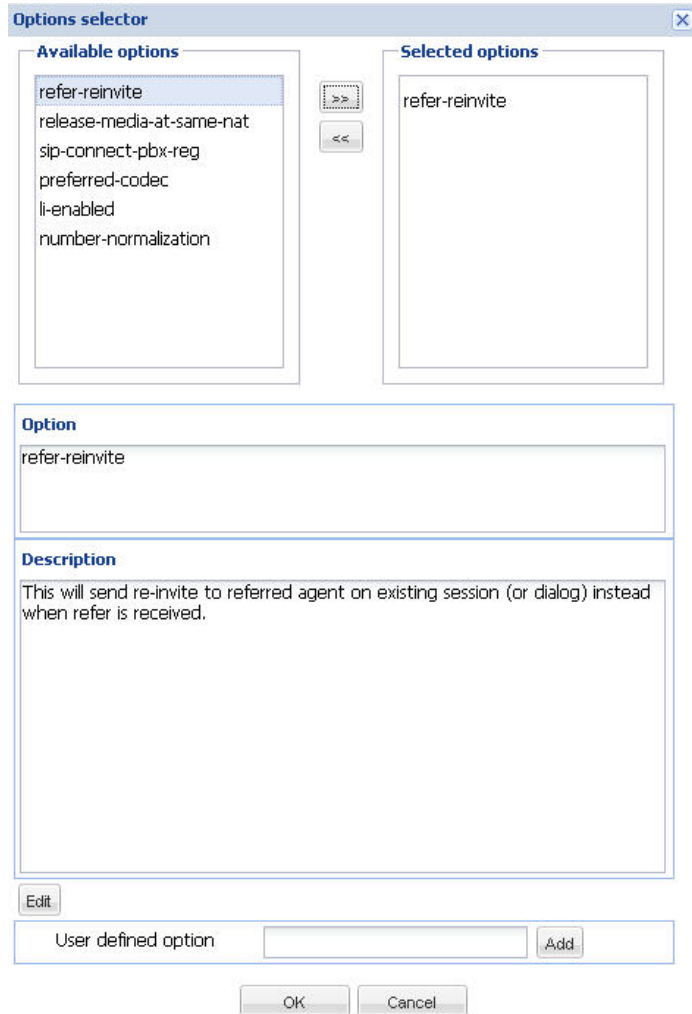
To configure SIP REFER with Replaces for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the realm table and click **Edit**.
4. Click the Realm tab and scroll to Options.
5. Click **Options**. The Options selector window appears.
6. Click **refer-reinvite** in the Available options list to select it.

7. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



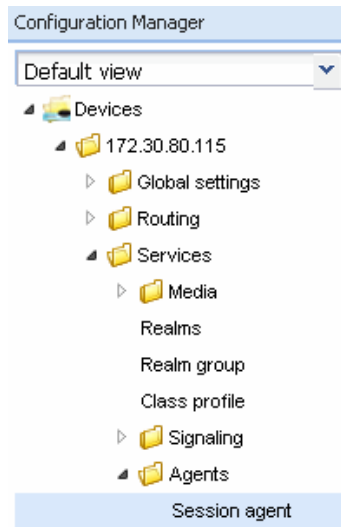
The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** list contains: refer-reinvite, release-media-at-same-nat, sip-connect-pbx-reg, preferred-codec, li-enabled, and number-normalization. The **Selected options** list contains: refer-reinvite. Between these lists are two buttons:  and . Below these lists is a detailed view for the selected option, **refer-reinvite**. It has an **Option** field containing 'refer-reinvite' and a **Description** field containing 'This will send re-invite to referred agent on existing session (or dialog) instead when refer is received.' At the bottom, there is an **Edit** button, a **User defined option** field, an **Add** button, and **OK** and **Cancel** buttons.

8. Click **OK**.
9. Click **Apply**.

Session Agents

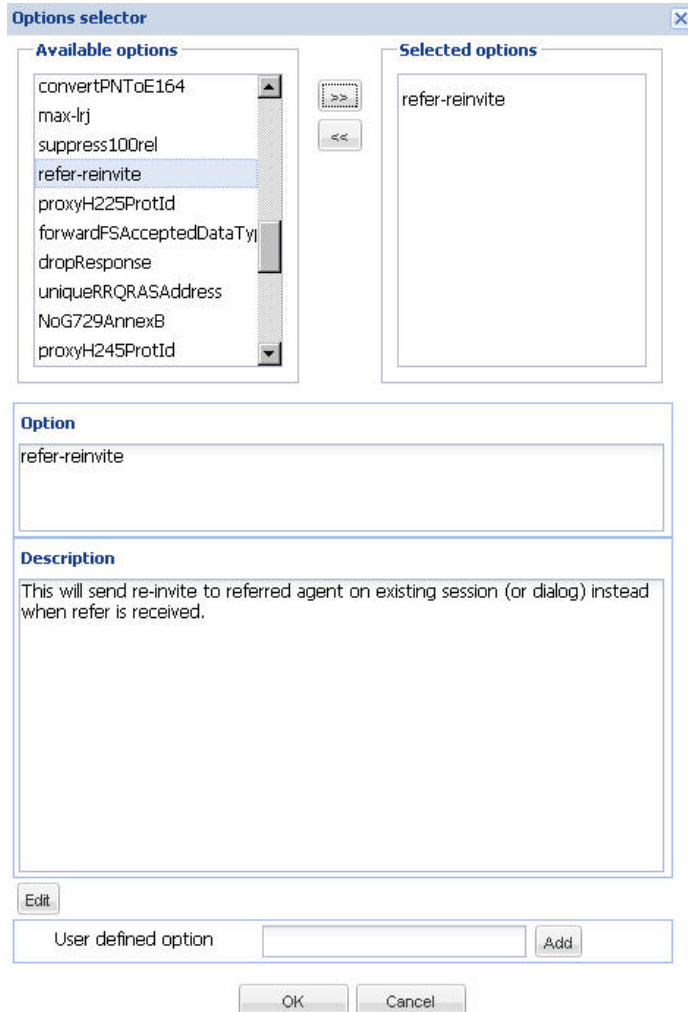
To configure SIP REFER with Replaces for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent you want to configure in the realm table and click **Edit**.
4. Click the Session agent tab and scroll to SIP options for this SA.
5. Click **Options**. The Options window appears.
6. Click **refer-reinvite** in the Available options list to select it.

7. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has two main panes: **Available options** and **Selected options**. In the **Available options** list, 'refer-reinvite' is selected. Between the panes are two arrow buttons: a right arrow (>>) and a left arrow (<<). The **Selected options** pane contains 'refer-reinvite'. Below the panes, there is a detailed view for the selected option:

Option
refer-reinvite

Description
This will send re-invite to referred agent on existing session (or dialog) instead when refer is received.

At the bottom, there is an **Edit** button, a text field labeled 'User defined option', an **Add** button, and **OK** and **Cancel** buttons.

8. Click **OK**.
9. Click **Apply**.

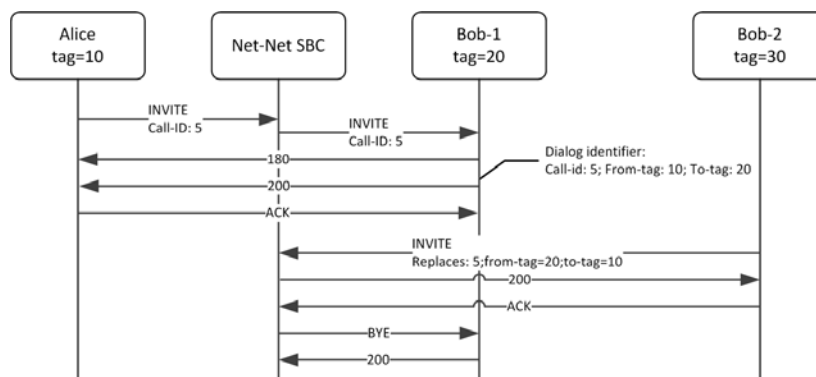
Replaces Header Support

The Net-Net DD supports the Replaces: header in SIP messages according to RFC 3891. The header, included within SIP INVITE messages, provides a mechanism to replace an existing early or established dialog with a different dialog which can be used for services such as call parking, attended call transfer and various conferencing features.

The Replaces: header indicates the dialog it wishes to replace by containing the corresponding dialog identifier. The identifier includes the triplet of the from tag, to tag, and call id. The orientation of endpoint-created tags, as from-tag and to-tag will match each of the two dialogs for a standard call. Thus the Replaces: header from an endpoint that indicates it wants to assume the dialog between the Net-Net DD and Bob-1 appears as:

Replaces:5555;from-tag=20;to-tag=10

The Net-Net DD validates that dialog identifier by matching an existing dialog and tries to install the new endpoint and remove the old endpoint by gracefully ending that dialog with a BYE. The replaces INVITE must come from an endpoint in the same realm as the endpoint it is replacing. If the UA sending the Replaces header is in a different realm as the original call leg (or indicates such architecture via a malformed Replaces: header), the Net-Net DD replies to the Replaces: endpoint with a 481 Missing Dialog. Refer to the following diagram for the standard case.



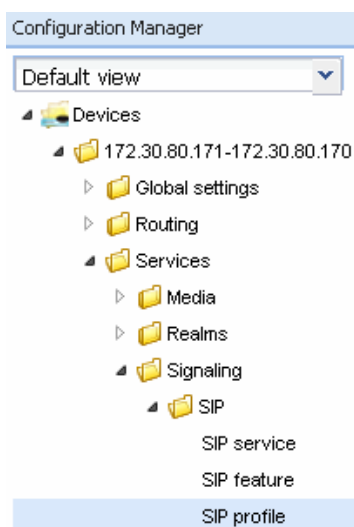
Configuration Instructions

Replaces: header support is configured in the session agent, realm, or SIP interface via their configuration elements SIP profile. SIP profiles are defined once and attached to a chosen interface, realm or session agent.

The replace-dialogs parameter is set to either **enabled** or **disabled**. In addition, you may set this parameter to **inherit** which uses the next lower order of precedence object. If there are no SIP profiles referenced in the higher ordered object, or if all the Replace dialogs parameters are set to **inherit**, then the feature is disabled.

To configure Replaces: header support:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.
3. Select a Sip profile from the table and click **Edit**.

4. **Replace dialogs**—Set this parameter to **enabled** to enable Replaces: header support. A replaces parameter is also inservtd in the Supported: header as sent into the realm where this SIP profile is applied. You may also set this parameter to **inherit** for the element which this SIP profile is applied to, to inheret the value from the next-lower element.
5. Click **Apply**.

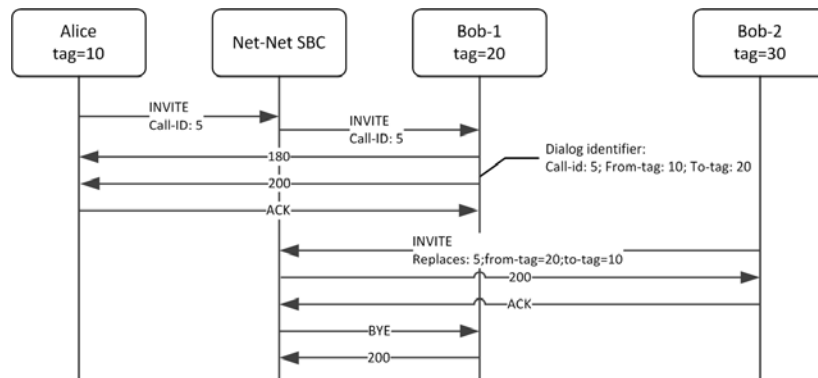
Replaces Header Support

The Net-Net SBC supports the Replaces: header in SIP messages according to RFC 3891. The header, included within SIP INVITE messages, provides a mechanism to replace an existing early or established dialog with a different dialog which can be used for services such as call parking, attended call transfer and various conferencing features.

The Replaces: header indicates the dialog it wishes to replace by containing the corresponding dialog identifier. The identifier includes the triplet of the from tag, to tag, and call id. The orientation of endpoint-created tags, as from-tag and to-tag will match each of the two dialogs for a standard call. Thus the Replaces: header from an endpoint that indicates it wants to assume the dialog between the Net-Net SBC and Bob-1 appears as:

Replaces:5555;from-tag=20;to-tag=10

The Net-Net SBC validates that dialog identifier by matching an existing dialog and tries to install the new endpoint and remove the old endpoint by gracefully ending that dialog with a BYE. The replaces INVITE must come from an edpoint in the same realm as the endpoint it is replacing. If the UA sending the Replaces header is in a different realm as the original call leg (or indicates such architecture via a malformed Replaces: header), the Net-Net SBC replies to the Replaces: endpoint with a 481 Missing Dialog. Refer to the following diagram for the standard case.



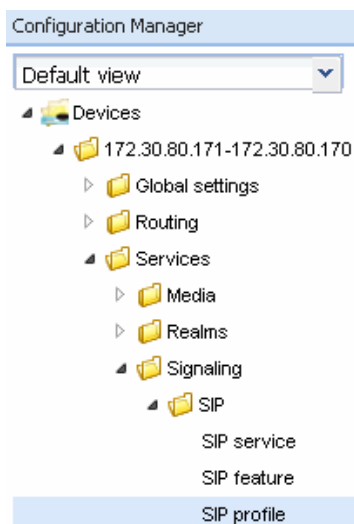
Configuration Instructions

Replaces: header support is configured in the session agent, realm, or SIP interface via their configuration elements SIP profile. SIP profiles are defined once and attached to a chosen interface, realm or session agent.

The replace-dialogs parameter is set to either **enabled** or **disabled**. In addition, you may set this parameter to **inherit** which uses the next lower order of precedence object. If there are no SIP profiles referenced in the higher ordered object, or if all the Replace dialogs parameters are set to **inherit**, then the feature is disabled.

To configure Replaces: header support:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.
3. Select a Sip profile from the table and click **Edit**.
4. **Replace dialogs**—Set this parameter to **enabled** to enable Replaces: header support. A replaces parameter is also insrvted in the Supported: header as sent into the realm where this SIP profile is applied. You may also set this parameter to **inherit** for the element which this SIP profile is applied to, to inheret the value from the next-lower element.
5. Click **Apply**.

SIP Session Timer Feature

SIP does not have a keepalive mechanism for established sessions and it does not have the capability of determining whether a session is still active. User agents (UAs) may be able to determine whether a session has timed out by using session specific mechanisms, but proxies cannot always determine when sessions are still active.

The Net-Net SBC provides a SIP session timer feature that, when enabled, forwards the re-INVITE or UPDATE requests from a User Agent Client (UAC) to a User Agent Server (UAS) in order to determine whether or not a session is still active. This refresh feature works for both UAs and proxies. The following paragraphs provide additional information about the session timers on the Net-Net SBC.

How the Session Timer Feature Works

During an active SIP call session, when a UA fails to send a BYE message at the end of the session, or when the BYE message gets lost due to network problems, the proxy cannot determine when the session has ended. Therefore, the proxy may hold onto resources associated with the call session for indefinite periods of time causing the session to never time out.

The SIP session timer feature adds the capability to periodically refresh SIP sessions

by sending repeated INVITE (re-INVITE) or UPDATE Session Refresh Requests. These requests are sent during active call legs to allow UAs or proxies to determine the status of a SIP session. The Session Refresh Requests along with the session timers determine if the active sessions stay active and completed sessions are terminated.

When you enable the session timer feature on the Net-Net SBC, the Session Director (SD) periodically sends out a Session Refresh Request (re-INVITE or UPDATE). The Response that is returned to the SD contains a success status code (2xx) that contains a session timer interval. The SD then refreshes the session timer each time it receives the 2xx Response containing that session timer interval.

The initial INVITE message sent from the UAC to the UAS contains two fields that make up the session timer interval in the SIP Session Header:

- Session-Expires (SE) - Specifies the maximum amount of time, in seconds, that can occur between session refresh requests in a dialog before the session is considered timed out.
- Minimum-SE (Min-SE) - Specifies the minimum allowed value, in seconds, for the session expiration.

The following displays the session timer interval values inserted in the SIP session INVITE message per RFC 4028:

```

INVITE sip:9109621001@192.168.200.99 SIP/2.0
Via: SIP/2.0/UDP
192.168.200.49:5060;branch=z9hG4bK0g6t23200gd0res4l580.1
Max-Forwards: 69
From: <sip:rick@192.168.1.48>;tag=SDr08od01-188c3fbc-b01a-4d68-
b741-09e5dc98a064
To: sip:149@192.168.1.49
Contact: <sip:rick@192.168.200.49:5060;transport=udp>
Call-ID: SDr08od01-9c12b48e3b0f7fad39ff3a2e0ced5ed3-v3000i1
CSeq: 3941 INVITE
Allow: INVITE, ACK, BYE, CANCEL, UPDATE, PRACK
Supported: timer
Session-Expires: 1800
Min-SE: 90
Content-Type: application/sdp
Content-Length: 236

v=0
o=- 3462189550 3462189550 IN IP4 192.168.200.49
s=pjmedia
c=IN IP4 192.168.200.49
t=0 0
m=audio 20000 RTP/AVP 0 8 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=sendrecv
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15

```

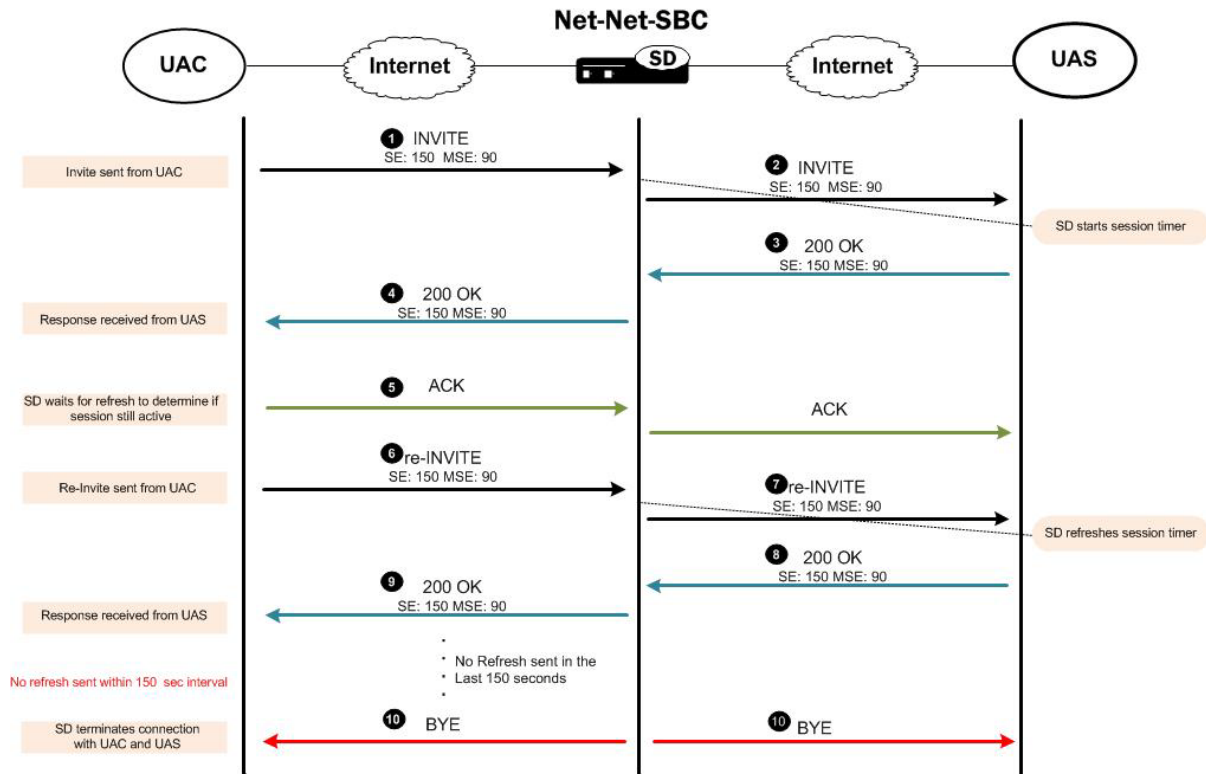
If the SD receives an INVITE from the UAC with a Session-Expires header, it starts a new session timer, or refreshes an existing session timer and then forwards the INVITE to the UAS. The subsequent 2xx Responses and re-INVITES also include the session timer intervals. If the SD does not receive a session refresh within the time specified in the session timer interval, the session timer expires, and the SD terminates the session between the UAC and the UAS.

The following occurs when you enable the session timer feature on the Net-Net SBC:

6. The UAC sends an INVITE to the SD with the SE and min-SE values (session timer interval). The SD starts a session timer.
7. The SD forwards the INVITE to the User Agent Server (UAS) with the same values.
8. The UAS sends the 200 OK Response to the SD with the session interval values.
9. The SD forwards the Response to the UAC.
10. The UAC sends an ACK (Acknowledge) to the SD and the SD forwards the ACK to the UAS.
11. The UAC sends out a re-INVITE (Session Refresh Request) to the SD with the session interval values. The SD refreshes the session timer.
12. The SD forwards the re-INVITE to the UAS.
13. The UAS sends the 200 OK response to the SD with the session interval values.

14. The SD sends the 200 OK response to the UAC.
15. If the SD does not receive a Response within the session interval (150 seconds in the following illustration), the timer expires, and the SD terminates the session between the UAC and the UAS.

The following illustration shows an example of a dialog between the UAC, the SD, and the UAS during an active session.



When the SD terminates a session it sends a BYE to both the ingress and egress call legs. If accounting is configured, the SD also sends a RADIUS stop record with "Acct-Terminate-Cause = Session-Timeout." For more information about this RADIUS termination cause, see [Session-Timeout RADIUS Value \(765\)](#).

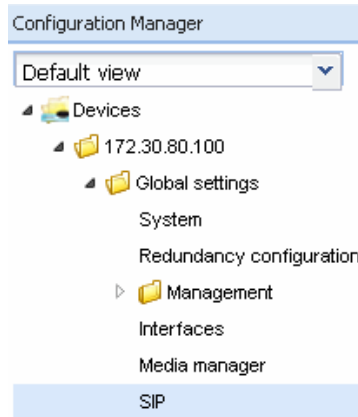
You can enable or disable the use of the session timers using the ACLI interface at `session-router > sip-config > options`.

Enabling the Session Timer Feature

You can configure the session timer feature on the Net-Net SBC to periodically refresh SIP sessions and determine whether or not a session is still active. If the Session Director (SD) determines that a session is no longer active, it terminates the session based on the session timer interval settings.

To configure the session timer feature on the Net-Net SBC:

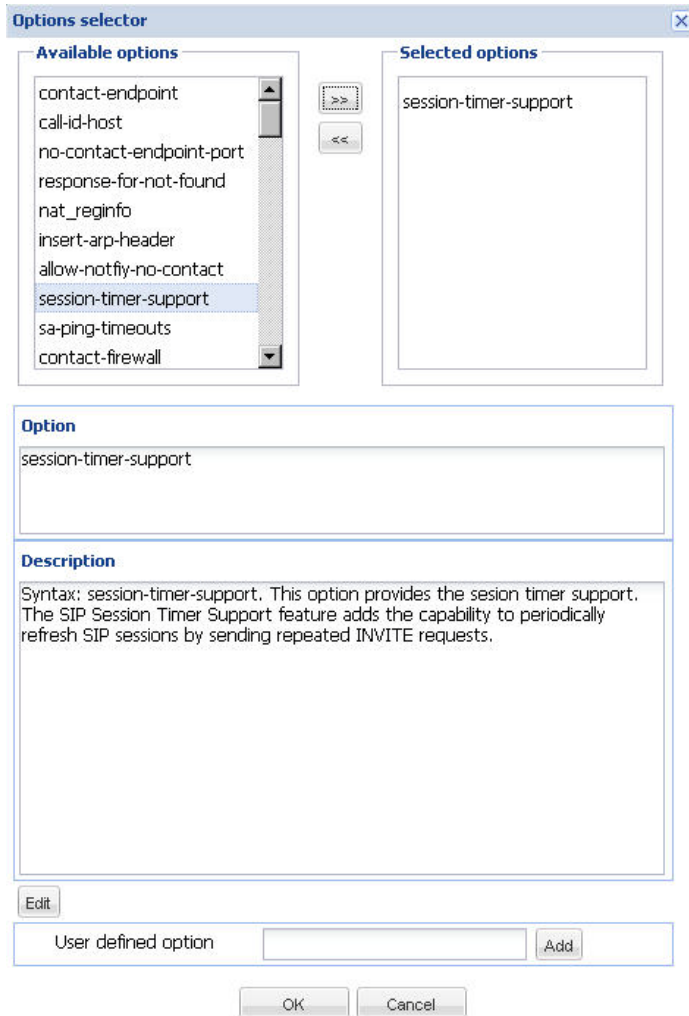
1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click the System folder.
5. Scroll to Options and click **Options**.
6. The Options selector window appears.
7. Click **session-timer-support** in the Available options list to select it.

This option enables the SD to start the session timer for session refreshes coming from the UAC. The SD determines whether or not a session is active based on session refreshes or responses. It terminates the session when no session refreshes occur within the session timer interval.

8. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main panes: **Available options** on the left and **Selected options** on the right. The **Available options** pane contains a list of options: contact-endpoint, call-id-host, no-contact-endpoint-port, response-for-not-found, nat_reginfo, insert-arp-header, allow-notfy-no-contact, session-timer-support (which is highlighted), sa-ping-timeouts, and contact-firewall. Between the panes are two buttons: a right arrow (>) and a left arrow (<). The **Selected options** pane contains the text 'session-timer-support'. Below the panes, there is a section for the selected option. It has a label **Option** followed by a text box containing 'session-timer-support'. Below that is a label **Description** followed by a larger text box containing the text: 'Syntax: session-timer-support. This option provides the sesion timer support. The SIP Session Timer Support feature adds the capability to periodically refresh SIP sessions by sending repeated INVITE requests.' At the bottom of the dialog, there is an **Edit** button, a text box labeled 'User defined option', and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

9. Click **OK**.
10. Click **Apply**.

Session-Timeout RADIUS Value

The Net-Net SBC supports the use of a RADIUS server for performing accounting, authentication, and authorization (AAA) tasks in the network.

If accounting is enabled on the Net-Net SBC, the “Acct-Terminate-Cause (49)” attribute is sent to the RADIUS server with a specific value indicating the reason for the account termination. Reasons for termination can be:

- User Request
- User Error
- Lost Service
- Admin Reset
- Admin Reboot
- NAS Request
- **Session Timeout**

If session timers are enabled, the “Acct-Terminate-Cause = “Session-Timeout” message is sent to the RADIUS server indicating the SD terminated a session on both the ingress and egress call legs due to inactivity in the session. No more Session Refresh Requests were sent in the session, so the SD terminated the session based on the session timer settings.

When session timers are disabled on the Net-Net SBC, no “Acct-Terminate-Cause = “Session-Timeout” is sent to the RADIUS server.

SIP Header Pre-Processing HMR

By default, the Net-Net SBC performs in-bound SIP manipulations after it carries out header validation. Adding the **inmanip-before-validate** option in the global SIP configuration allows the Net-Net SBC to perform HMR on received requests prior to header validation. Because there are occasional issues with other SIP implementations—causing invalid headers to be used in messages they send to the Net-Net SBC—it can be beneficial to use HMR to remove or repair these faulty headers before the request bearing them are rejected.

When configured to do so, the Net-Net SBC performs pre-validation header manipulation immediately after it executes the top via check. Inbound SIP manipulations are performed in order of increasing precedence: SIP interface, realm, and session agent.

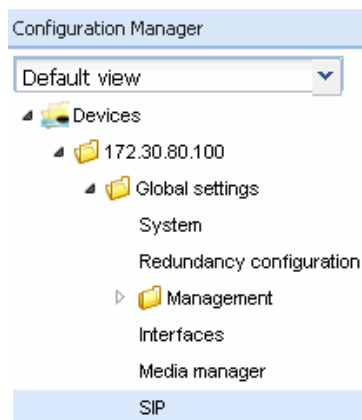
The fact that the top via check happens right before the Net-Net SBC carries out pre-validation header manipulations means that you cannot use this capability to repairs the first via header if it is indeed invalid. If pre-validation header manipulation were to take place at another time during processing, it would not be possible to use it for SIP session agents. The system learns of matching session agents after top via checking completes.

For logistical reasons, this capability does not extend to SIP responses. Inbound manipulation for responses cannot be performed any sooner that it does by default, a time already preceding any header validation.

Enabling SIP Header Pre-Processing

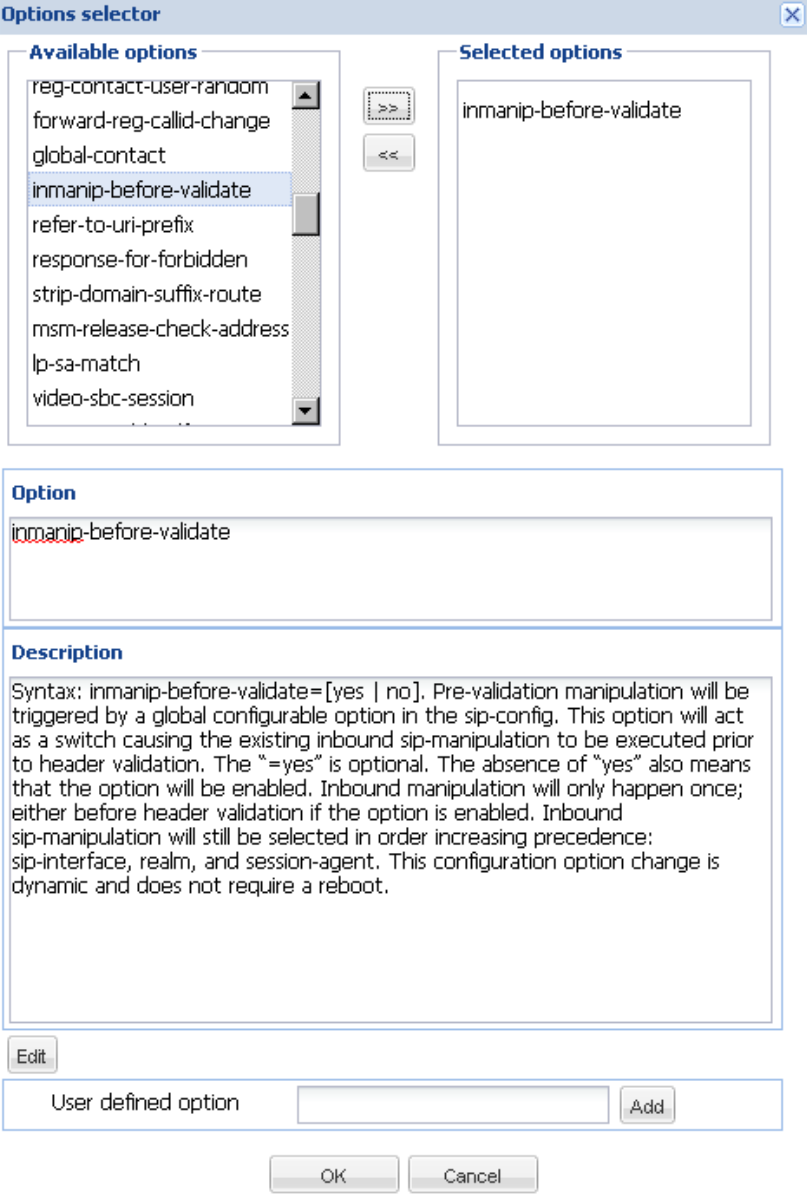
To enable SIP header pre-processing:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click the System tab.
5. Scroll to Options.
6. Click **Options**.
7. Click **inmanip-before-validate** in the Available options list to select it.

8. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** section contains a list box with the following items: reg-contact-user-random, forward-reg-callid-change, global-contact, inmanip-before-validate (which is highlighted), refer-to-uri-prefix, response-for-forbidden, strip-domain-suffix-route, msm-release-check-address, lp-sa-match, and video-sbc-session. Between the two list boxes are two buttons: a right arrow (>) and a left arrow (<). The **Selected options** section contains a text area with the text "inmanip-before-validate". Below these sections is a detailed view for the selected option. It has a label **Option** followed by the text "inmanip-before-validate". Below that is a label **Description** followed by a text area containing the following text: "Syntax: inmanip-before-validate=[yes | no]. Pre-validation manipulation will be triggered by a global configurable option in the sip-config. This option will act as a switch causing the existing inbound sip-manipulation to be executed prior to header validation. The "=yes" is optional. The absence of "yes" also means that the option will be enabled. Inbound manipulation will only happen once; either before header validation if the option is enabled. Inbound sip-manipulation will still be selected in order increasing precedence: sip-interface, realm, and session-agent. This configuration option change is dynamic and does not require a reboot." Below the description is an **Edit** button. At the bottom of the dialog is a section for **User defined option** with a text input field and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

9. Click **OK**.
10. Click **Apply**.

LP Routing with ENUM provided SA Route

You can configure the Net-Net SBC to match to Net-Net 3000 and 4000 Series behavior with regard to a session agent route as chosen by a local policy match, returned from an ENUM query.

When the Net-Net 9200 matches a session agent route from a local policy ENUM query, the systems forward the message toward the chosen session agent with two restrictions:

1. The Net-Net SBC does not enforce session agent constraints and collected statistics for flows between the system and the chosen session agent.
2. If no realm is configured in the matching policy attribute, the traffic is not forwarded into the session agent's realm (configured in the session agent configuration element), but to the sip-config's egress realm.

To override this behavior and configure the Net-Net 9200 to mimic Net-Net 3000 and 4000 series behavior, you must set the **ENUM session agent match** parameter to **enabled**. Doing so will cause the Net-Net 9200 to:

1. Enforce session agent constraints and collected statistics for flows between the system and the chosen session agent.
2. Forward traffic to the chosen session agent's realm (in the session agent configuration element), if the chosen policy attribute's realm parameter is blank. If these first two realms are unknown (policy attribute and session agent), the Net-Net 9200 sends the traffic into the sip-config's realm.

All local policy queries that return session agent targets are matched based on the algorithm described in the Local Policy Session Agent Matching for SIP in the SIP Services chapter.

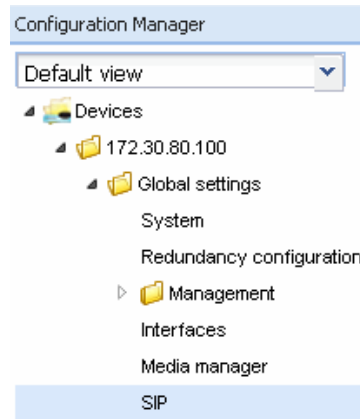
Upgrades

Because enabling this behavior has dramatic effects on routing behavior, this parameter's value defaults to disabled when upgrading to a release with **ENUM session agent match** available for the first time. When creating a sip-config configuration element for the first time, the **ENUM agent match** parameter will be set to enabled.

Configuring Local Policy Routing with ENUM Provided Session Agents

To configure local policy routing with ENUM provided session agents:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **ENUM agent match**—Select **enabled** from the drop-down list to apply session agent constraints, statistics, and routing functionality when using routes returned from an ENUM query.

ENUM agent match

5. Click **Apply**.

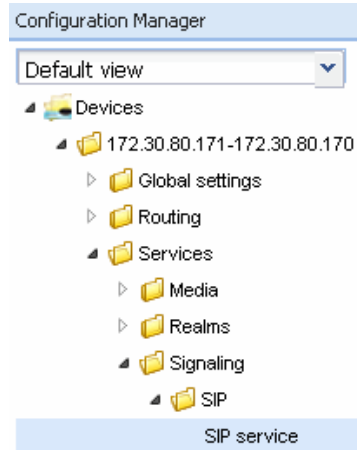
Simultaneous TCP Connection and Registration Cache Deletion

You can now configure simultaneous TCP connections and registration cache deletion.

Configuring Simultaneous TCP Connection and Registration Cache Deletion

To configure simultaneous TCP connection and registration cache deletion:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP Interface tab.
4. Click the SIP interface you want to configure and click **Edit**.
5. Scroll to TCP/TLS Denial service protection.
6. **Number of seconds after de-registration to kill TCP connection**—Enter the number of seconds after which the TCP connection will be disabled. The valid range is 0-999999999. The default value is 0.

Number of seconds after de-registration to kill TCP connection

7. Click **Apply**.

STUN Keep-Alive

Release S-C6.2.1 provides an alternative NAT (Network Address Translator) Traversal method. The currently supported method is described in the *SIP Hosted-NAT Traversal (HNT)* section of the *SIP Signaling Services* chapter of the *Net-Net 4000 ACLI Configuration Guide*. The current method is SBC-based and requires no explicit participation by the SIP endpoint. Rather the SBC manipulates SIP registration requests and responses to spoof the endpoint — causing it to issue frequent and extraneous registration requests thus maintaining existing NAT bindings.

The alternative method is based upon RFC 5626, *Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)*, and a recent internet draft, *Indication of Support for Keep-Alive (draft-ietf-sipcore-keep-08.txt)*. Unlike the current SBC-centric method, the new alternative requires the active participation of the SIP endpoint. With this method the SIP endpoint and the SBC negotiate a

request/response message sequence which generates sufficient traffic flow to maintain NAT bindings.

Section 3.5.2 of RFC 5626 defines a keep-alive method for connectionless UDP connections., but provides no guidance for keep-alive negotiation. The *Indication of Support for Keep-Alive* internet draft addresses this deficiency by defining a procedure that enables a SIP endpoint to signal its capability and willingness to send and receive periodic keep-alive messages to a device referred to by the draft as an *edge proxy*, a role performed by the SBC. After receiving such a signal, the SBC returns a response indicating its willingness to exchange keep-alives, and specifying the frequency of the exchange.

Refer to How It Works for operational and procedural details.

Platform Support

STUN Keep-Alive is supported on the following platforms: NN4250, and NN4500.

License Requirements

STUN Keep-Alive support requires no special license.

How It Works

SIP endpoints that initiate and participate in the keep-alive exchanges described in this section must support a minimal sub-set of client operations as described in RFC 5389, *Session Traversal Utilities for NAT (STUN)*. Specifically, endpoints must be able to construct and transmit STUN binding requests, and receive and parse STUN binding responses. Binding request and response formats are described in Section 6 of RFC 5389.

As shown in the following SIP Registration request, the SIP endpoint, functioning as a STUN client, signals its willingness to exchange keep-alive messages by placing an unvalued *keep* parameter (newly-defined by the *Indication of Support for Keep-Alive* internet draft) in the SIP *Via* header. The *expires* parameter in the *Contact* header requests a registration period of 5 hours (18000 seconds).

```
REGISTER sip:512@172.16.101.23:5060 SIP/2.0
Via: SIP/2.0/UDP 172.16.101.38:5070;branch=dd1;keep
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512" <sip:512@172.16.101.38:5070>
Call-ID:1-14400@172.16.101.38
CSeq: 1 REGISTER
Max-Forwards: 70
Contact: "512" <sip:512@172.16.101.38:5070>;expires:18000
User-Agent: ADTRAN_Total_Access_908e_(2nd_Gen)/A1.02.00.E
Content-Length: 0
```

The SBC forwards the Registration request (absent the *keep* parameter) to the Registrar.

```
REGISTER sip:512@192.168.7.32:5060 SIP/2.0
Via: SIP/2.0/UDP
192.168.101.23:5060;branch=z9hG4bK3q43klh3dafekdepnbaqip04e1
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512" <sip:512@172.16.101.38:5070>
Call-ID: 1-14400@172.16.101.38
CSeq: 1 REGISTER
```

```

Max-Forwards: 69
Contact: "512"
<sip:512@192.168.101.23:5060;transport=udp>;expires:18000
User-Agent: ADTRAN_Total_Access_908e_(2nd_Gen)/A1.02.00.E
Content-Length: 0
Allow: ACK, BYE, CANCEL, INFO, INVITE, NOTIFY, OPTIONS, REFER

```

The Registrar indicates successful registration with a 200 OK response back to the SBC. The *expires* parameter in the *Contact* header grants a registration period of 1 hour (3600 seconds).

```

SIP/2.0 200 OK
Via: SIP/2.0/UDP
192.168.101.23:5060;branch=z9hG4bK3q43klh3dafekdepnbaqip04e1
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512"
<sip:512@172.16.101.38:5070>;tag=b68b3d53a5a90225609112ff6c211bef.16a6
Call-ID: 1-14400@172.16.101.38
CSeq: 1 REGISTER
Contact: <sip:512@192.168.101.23:5060;transport=udp>;expires=3600
Server: OpenSER (1.3.0-notls (i386/linux))
Content-Length: 0

```

The SBC, acting as a STUN server, forwards the 200 OK to the endpoint after inserting a *keep* parameter and a parameter value in the *Via* header of the Registration response. The presence of the *keep* parameter signals the SBC's willingness to exchange keep-alives, and the parameter value specifies the exchange frequency in seconds.

```

SIP/2.0 200 OK
Via: SIP/2.0/UDP 172.16.101.38:5070;branch=dd1;keep=20
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512"
<sip:512@172.16.101.38:5070>;tag=b68b3d53a5a90225609112ff6c211bef.16a6
Call-ID: 1-14400@172.16.101.38
CSeq: 1 REGISTER
Contact: <sip:512@172.16.101.38:5070>;expires=3600
Server: OpenSER (1.3.0-notls (i386/linux))
Content-Length: 0

```

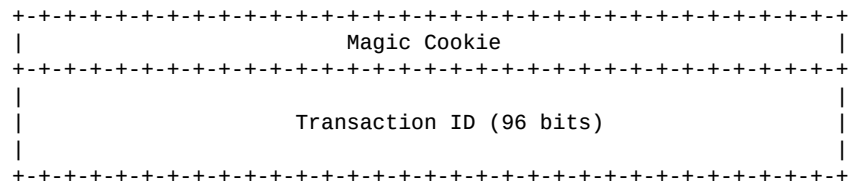
After the keep-alive exchange has been negotiated, the SIP endpoint, acting as a STUN client, is required to transmit a periodic STUN binding request so that the interval between each request is randomly distributed between 80 and 100 percent of the value of the *keep* parameter. Assuming a parameter value of 20 seconds, for example, the SIP endpoint transmits a STUN binding request at random intervals between 16 and 20 seconds in length.

The STUN binding request has the following format.

```

0           1           2           3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|0 0|          STUN Message Type          |          Message Length          |

```

The initial two bits of any STUN message are always 00

STUN Message Type

This 14-bit field contains the message class and method — for a STUN binding request, class is *request* and method is *binding*.

Message Length

This 16-bit field contains the length, in octets, of the Attribute section of the STUN binding request. Since the Attribute section is not present in a binding request, this field contains a value of 0.

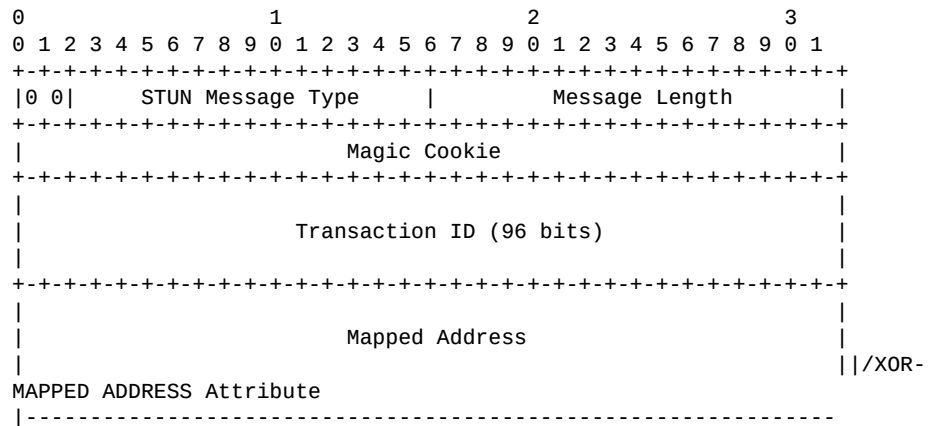
Magic Cookie

This 16-bit field always contains a value of 0x2112A442.

Transaction ID

This 96-bit field contains a random value that provides a transaction identifier. The value is generated by the STUN client and echoed by the STUN server.

Upon receipt of a binding request, the SBC, acting as a STUN server, transmits a STUN binding response. Receipt of the binding response by the endpoint confirms the UDP connection between the endpoint and the SBC, and the viability of NAT bindings in the transmission path. The STUN binding response has the following format.



The initial two bits of any STUN message are always 00.

STUN Message Type

This 14-bit field contains the message class and method — for a STUN binding response, class is *response* and method is *binding*.

Message Length

This 16-bit field contains the length, in octets, of the Attribute section of the STUN binding response. Since the binding response contains a single attribute,

MAPPED-ADDRESS/XOR-MAPPED-ADDRESS, the message length can be either 8 or 16 octets, depending on the IP address type.

Magic Cookie

This 16-bit field always contains a value of 0x2112A442.

Transaction ID

This 96-bit field contains a random value that provides a transaction identifier. The value is generated by the STUN client and echoed by the STUN server.

MAPPED-ADDRESS

This attribute contains the IP address and port number of the proximate NAT device, that is the address translator closest to the SBC, that forwarded the STUN request toward the SBC. Specific fields within the attribute identify the IP family (Version 4 or 6), the IP address, and port number. The attribute length is 8 octets if it contains an IPv4 address, or 16 octets if it contains an IPv6 address.

XOR-MAPPED-ADDRESS

This attribute contains the obfuscated IP address and port number of the proximate NAT device, that is the address translator closest to the SBC, that forwarded the STUN request toward the SBC. Obfuscation is accomplished by X-ORing the plaintext IP address and port values with the Magic Cookie and Transaction ID contents. Specific fields within the attribute identify the IP family (Version 4 or 6), the obfuscated IP address, and obfuscated port number. The attribute length is 8 octets if it contains an IPv4 address, or 16 octets if it contains an IPv6 address.

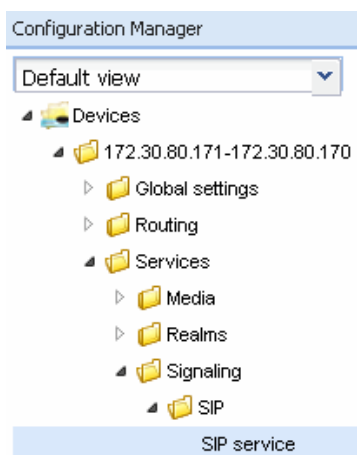
Once initiated, endpoint transmission of STUN binding requests and SBC responses continue for the duration of the SIP Registration, 1 hour in the above example, or until the endpoint transmits a new Registration request. In the event of such a request, the endpoint must once again indicate its willingness to exchange STUN keep-alives with an unvalued *keep* parameter in the *Via* header. If keep-alive renegotiation is not successful, the endpoint must cease the transmission of keep-alive messages.

An endpoint failure to issue a timely STUN binding request is not fatal. In the absence of an expected request, the SBC takes no action with regard to the UDP connection, or to established sessions.

Configuring STUN Keep-Alive

To configure STUN keepalive:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP Interface tab.
4. Click the SIP service you want to configure in the table and click **Edit**.
5. Click the Registration tab.
6. Scroll to Registration handling.
7. **Register keepalive**—Click a value in the drop down list. The valid values are:
 - *none* — (the default) disables STUN keep-alive
 - *always* — assuming that the endpoint has included the *keep* parameter in the *Via* header, exchange STUN keep-alives with that endpoint
 - *bnat* — assuming that the endpoint has included the *keep* parameter in the *Via* header, exchange STUN keep-alives only with endpoints that are behind an intervening NAT device (based on comparing source IP packet addresses with IP addresses extracted from the SIP request).

Registration keepalive bnat ▼

8. Click **Apply**.

P-Asserted-ID Output to CDRs

The Net-Net SBC does not write VSA 69, Acme-P-Asserted-ID, if the following two conditions are true:

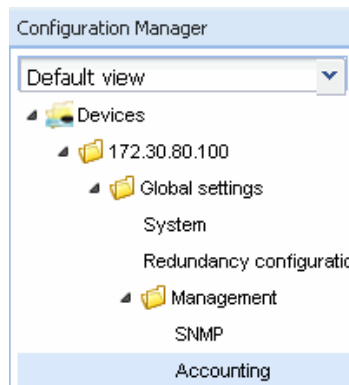
1. The SIP message's destination is untrusted
2. Privacy is on. i.e., an id is contained within the Privacy: header.

In order to override this behavior and write the P-Asserted-ID's header value to a CDR, enable the **insert-pai** option in the account config.

Configuring P-Asserted IDs in CDRs

To configure the Net-Net 9200 to include P-Asserted-IDs in CDRs:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Scroll to Options.
5. Click **Options**. The Options selector dialog box appears.
6. Click **insert-pai** in the Available options list to select it.
7. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.
8. Click **OK**.
9. Click **Apply**.

P-Asserted-ID Output to CDRs

The Net-Net SBC does not write VSA 69, Acme-P-Asserted-ID, if the following two conditions are true:

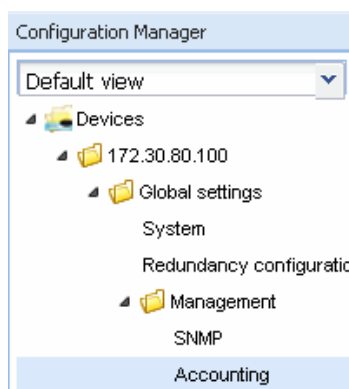
1. The SIP message's destination is untrusted
2. Privacy is on. i.e., an id is contained within the Privacy: header.

In order to override this behavior and write the P-Asserted-ID's header value to a CDR, enable the **insert-pai** option in the account config.

Configuring P-Asserted IDs in CDRs

To configure the Net-Net 9200 to include P-Asserted-IDs in CDRs:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. Scroll to Options and click **Options**.
5. Click **insert-pai** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

P-CSCF: SIP Registration Event Package

This section describes the P-CSCF SIP Registration Event Package functionality, used when the Net-Net SBC is used as a P-CSCF in IMS cores to provide a clear interface between the P-CSCF and the rest of the core. Within IMS cores, there are compliant and non-compliant wireless IMS clients. The Net-Net SBC supports implicit registration by allowing you to enable the RegEventPackage (as specified in 3GPP 24.229) on a SIP interface.

The Reg-Event-Package performs the role of allowing implicit registration. Without it, the P-Associated-URI or a similar header is used and only provides unreliable information because it does not contain the implicit registration set and does not communicate the IMPU's registration status.

How It Works

When it receives a 2xx response to an initial registration for a new contact, the Net-Net SBC subscribes to the registration event package for the public user identity (AOR) which is registered with the user's registrar (S-CSCF). The Net-Net SBC uses the default public user identity (i.e., the first URI in the list present in the P-Associated-URI) for subscription to the registration-state event package if the public user identity used at initial registration is a barred public user identity. If the public user identity is not barred, the Net-Net SBC uses either the default public user identity or the public user identity for initial registration.

Then Net-Net SBC sends a SUBSCRIBE request with header fields populated as follows:

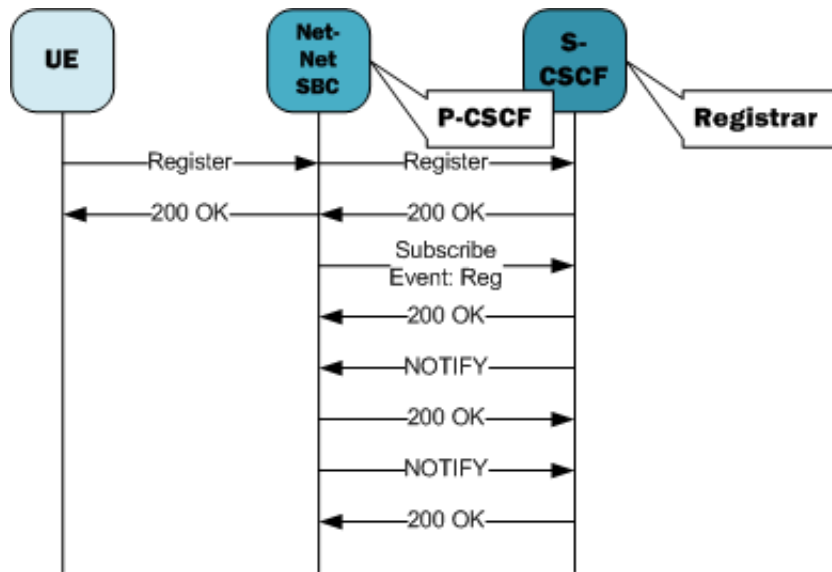
- Request-URI: Set to the resource to which the Net-Net SBC wishes to subscribe; i.e., a SIP URI containing the public user identity used for the subscription
- From: Set to a SIP URI containing the public user identity used for the subscription
- To: Set to a containing the public user identity used for the subscription
- Event: Set to the RegEventPackage
- Expires: Set to 600000 seconds, the desired value for the duration of the subscription
- P-Asserted-Identity: Set to the P-CSCF's SIP URI, which had been inserted into the Path header field during the registration of the user to whose user state the P-CSCF subscribes
- Contact: Set to contain the same IP address or FQDN found in the previous Register message sent to the Registrar; multiple registrations are supported
- Route: Set to the P-CSCF address learned during P-CSCF discovery and including the Service-Route header elements from registration; the P-CSCF URI contains the port number learned from the security negotiation

The following describes Net-Net SBC behaviors in different scenarios after it sends the SUBSCRIBE request:

- If it receives a 2xx response to the SUBSCRIBE, the Net-Net SBC maintains the generated dialog, which is identified by the values in the Call-ID header field and To and From header field tags.
- If continued subscription is required after it receives the 2xx response, the Net-Net SBC information stored for establishing the dialog and for setting the expiration time. It then automatically refreshes the subscription RegEventPackage for a registered public user identity in either of these ways:

- 600 seconds before the expiration time if the initial subscription was greater than 1200 seconds
- When half of the time has expired if the initial subscription was equal to or less than 1200 seconds
- When a SUBSCRIBE request to refresh a subscription fails with any response other than a 481, the Net-Net SBC deems the original subscription valid for the duration of the most recently known Expires value (according to RFC 3265). Otherwise, the Net-Net SBC invalidates the subscription and starts a new initial subscription.
- When a SUBSCRIBE request to refresh a subscription fails with a 481, the Net-Net SBC invalidates the subscription because that subscription has been terminated and the subscriber has not been notified of the termination.
- When it receives a 503 Service Unavailable response to an SUBSCRIBE with a Retry-After header field, the Net-Net SBC does not retry the request until the period indicated in this header field elapses.
- In the case of user-initiated deregistration, the Net-Net SBC sends a SUBSCRIBE request if all public user identities previously user-registered using a private user identity have been deregistered. This SUBSCRIBE has an Expires header set to zero (0); afterwards, the Net-Net SBC receives a NOTIFY from the S-CSCF in which the subscription state is terminated. If the subscription state is not set to terminated, the Net-Net SBC can either unsubscribe to the user's RegEventPackage or let the subscription expire.
- In the case of network-initiated deregistration, the Net-Net SBC removes all registration details for the public user entities. This happens when a NOTIFY request on any dialog generated during the subscription to the RegEventPackage was registered to the Net-Net SBC with:
 - The state attribute is set to "terminated" and the event attribute within the Contact element for this UE is set to "rejected" or "deactivated"
 - The state attribute is set to "active" and in the Contact element for this UE, the state attribute is set to "terminated" and the associated event attribute is set to "rejected" or "deactivated"
- When it receives a NOTIFY request generated during the subscription to the RegEventPackage, the Net-Net SBC:
 - Stores the indicated public user identities as registered if the state attribute is set to "active"
 - Stores the indicated public user identities as deregistered if the state attribute is set to "terminated"

The diagram below shows the basic flow:



RegInfo Handling

The Net-Net SBC supports the RegInfo structure the NOTIFY messages it receives.

For the Registration element, which can contain multiple Contact sub-elements each with information for a single contact:

- AoR: A URI that is the address of record to which this registration refers.
- ID: Identifies this registration.
- State: State of the registration; “init” (creates cache entry for the AoR if none is present), “active” (sets the AoR cache entry to valid, i.e. active) or “terminated” (sets the AoR cache entry to invalid and removes it); the Net-Net SBC updates the cache entry when it receives new state information

For the Contact element, which contains a URI element and optionally a display-name element; the Net-Net SBC ignores other optional elements:

- ID: Identifies this contact.
- State: State of the registration; “active” (sets binding to valid in cache entry) and “terminated” (sets binding to invalid in cache entry)
- Event: Sets the current state of the contact state machine; registered, created, refreshed, shortened, expired, deactivated, probation, unregistered, or rejected

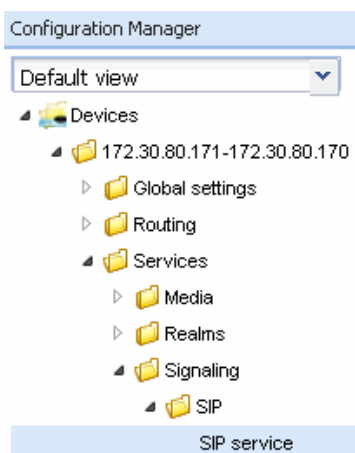
Consideration for HA Nodes

If you are using this RegEventPackage support in an HA node, the Net-Net SBC proceeds with the SUBSCRIBE reg-event procedure on a newly-active system although it had been started on the previously-active system.

Enabling P-CSCF RegEventPackage

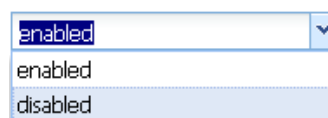
To enable the P-CSCF SIP RegEventPackage for a SIP interface:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP interface tab.
4. Click the SIP interface you want to configure in the table and click **Edit**.
5. Click the General tab.
6. **Subscribe reg event**—To enable support, click **enabled** in the drop down list. The default value is disabled.

Subscribe reg event



7. Click **Apply**.

2423 - SIP-KPML to RFC 2833 Conversion for DTMF Events

The Net-Net SBC supports SIP KPML to RFC 2833 conversion for DTMF events.

SIP KPML to RFC 2833 Negotiation

The Net-Net SBC removes the Allow-Event header if it contains the value “KPML”. A SUBSCRIBE request is sent for each KPML event upon a successful invite.

After the SUBSCRIBE even is successful, the Net-Net SBC will process all NOTIFY requests and respond with 200 OK messages.

The SDP-Insertion feature will be used if the Access-side does not offer a SDP in the invite and the core side does not accept an offerless invite.

RFC 2833 to SIP KPML Negotiation

The Net-Net SBC adds an Allow-Event header with the value “KPML” to the INVITE message.

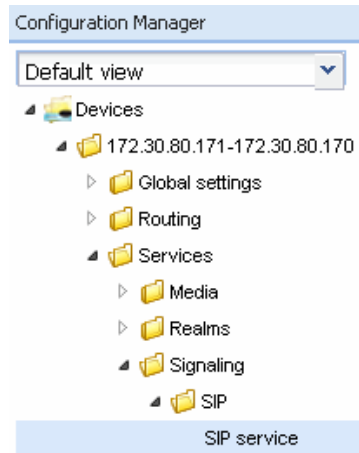
If a SUBSCRIBE message is recieved from the Access side, the Net-Net SBC will respond with a 200 OK message.

A KPML NOTIFY request is sent for each 2833 DTMF entered, until the subscription is terminated.

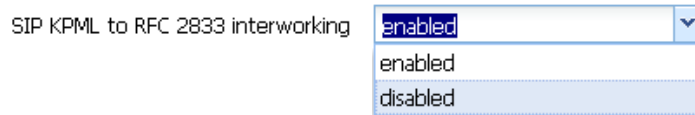
Configuring KPML—2833 Interworking

To configure KPML—2833 interworking:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP interface tab.
4. Click the SIP interface you want to configure in the table and click **Edit**.
5. **SIP KPML to RFC 2833 interworking**—Click **enabled** in the drop down list to use KPML-2833 interworking. The default value is **disabled**.



6. Click **Apply**.

SIP Subscriptions Management

There are new global and realm-specific SIP subscription counters.

Global Subscriptions Counters

Four 32-bit counters record system-wide subscription data.

apSysMgmtGeneralObjects Table (1.3.6.1.4.1.9148.3.2.1.1)

Object Name	Object OID	Description
apSysSipStatsActiveSubscriptions	apSysMgmtGeneralObjects: 1.3.6.1.4.1.9148.3.2.1.1.27	An unsigned 32-bit integer that specifies the current global count of active SIP subscriptions.
apSysSipStatsPerMaxSubscriptions	apSysMgmtGeneralObjects: 1.3.6.1.4.1.9148.3.2.1.1.28	An unsigned 32-bit integer that specifies the maximum global count of SIP subscriptions initiated during any 100 second period since the last SBC re-boot.
apSysSipStatsPerMaximumActiveSubscriptions	apSysMgmtGeneralObjects: 1.3.6.1.4.1.9148.3.2.1.1.29	An unsigned 32-bit integer that specifies the maximum global count of active SIP subscriptions since the last SBC re-boot.
apSysSipStatsTotalSubscriptions	apSysMgmtGeneralObjects: 1.3.6.1.4.1.9148.3.2.1.1.30	An unsigned 32-bit integer that specifies the global count of active SIP subscriptions since the last SBC re-boot.

Realm-Specific Subscriptions Counters

Four 32-bit counters record realm-based subscription data.

apSigRealmStatsEntry Table (1.3.6.1.4.1.9148.3.2.1.2.4.1)

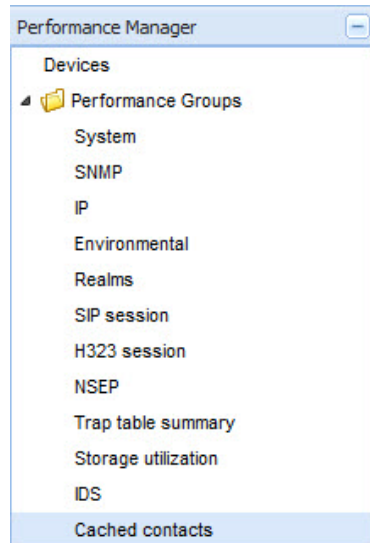
Object Name	Object OID	Description
apSigRealmStatsActiveSubscriptions	apSigRealmStatsEntry: 1.3.6.1.4.1.9148.3.2.1.2.4.1.32	An unsigned 32-bit integer that specifies the current domain count of active SIP subscriptions.
apSigRealmStatsPerMaxSubscriptions	apSigRealmStatsEntry: 1.3.6.1.4.1.9148.3.2.1.2.4.1.33	An unsigned 32-bit integer that specifies the maximum domain count of SIP subscriptions initiated during any 100 second period since the last SBC re-boot.

apSigRealmStatsEntry Table (1.3.6.1.4.1.9148.3.2.1.2.4.1)

Object Name	Object OID	Description
apSigRealmStatsMaximumActiveSubscriptions	apSigRealmStatsEntry: 1.3.6.1.4.1.9148.3.2.1.2.4.1.34	An unsigned 32-bit integer that specifies the maximum domain count of active SIP subscriptions since the last SBC re-boot.
apSigRealmStatsTotalSubscriptions	apSigRealmStatsEntry: 1.3.6.1.4.1.9148.3.2.1.2.4.1.35	An unsigned 32-bit integer that specifies the total domain count of active SIP subscriptions since the last SBC re-boot.

To access SIP subscriptions management statistics:

1. Expand the Performance Manager slider.
2. Select the SBC system you would like to poll and click **View**.
3. Click **Cached Contacts**.. The statistics appear in the main window.



SIP local contacts	0
MCGP GW endpoints	0
H323 registrations	0
Active SIP subscriptions	0
Maximum SIP subscriptions	0
Maximum active SIP subscriptions	0
Total SIP subscriptions	0

Keep-Alive with CR/LF

Release S-C6.2.1 provides an alternative NAT (Network Address Translator) Traversal method. The currently supported method is described in the *SIP Hosted-NAT Traversal (HNT)* section of the *SIP Signaling Services* chapter of the *Net-Net 4000 ACLI Configuration Guide*. The current method is SBC-based and requires no explicit participation by the SIP endpoint. Rather the SBC manipulates SIP registration requests and responses to the endpoint — causing it to issue frequent and extraneous registration requests thus maintaining existing NAT bindings.

The alternative method is based upon RFC 5626, *Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)*, and RFC 6223. Unlike the current SBC-centric method, the new alternative requires the active participation of the SIP endpoint. With this method the SIP endpoint and the SBC negotiate a request / response message sequence which generates sufficient traffic flow to maintain NAT bindings.

Section 3.5.2 of RFC 5626 defines a keep-alive method for connectionless UDP flows, but provides no guidance for keep-alive negotiation. The *Indication of Support for Keep-Alive* internet draft addresses this deficiency by defining a procedure that enables a SIP endpoint to signal its capability and willingness to send and receive periodic keep-alive messages to a device referred to by the RFC as an *edge proxy*, a role performed by the SBC. After receiving such a signal, the SBC returns a response indicating its willingness to exchange keep-alives, and specifying the frequency of the exchange.

Refer to [How It Works](#) for operational and procedural details.

Availability

N/A

Platform Support

CR/LF Keep-Alive is supported on the following platform: NN4500.

License Requirements

CR/LF Keep-Alive support requires no special license.

How It Works

SIP endpoints that initiate and participate in the keep-alive exchanges described in this section must support a minimal sub-set of client operations. Specifically, endpoints must be able to construct and transmit CR/LF binding requests, and receive and parse CR/LF binding responses. Binding request and response formats are described in Section 6 of RFC 5626.

As shown in the following SIP Registration request, the SIP endpoint, functioning as a CR/LF client, signals its willingness to exchange keep-alive messages by placing an unvalued *keep* parameter, newly-defined by the [RFC] 6223 in the SIP *Via* header. The *expires* parameter in the *Contact* header requests a registration period of 5 hours (18000 seconds).

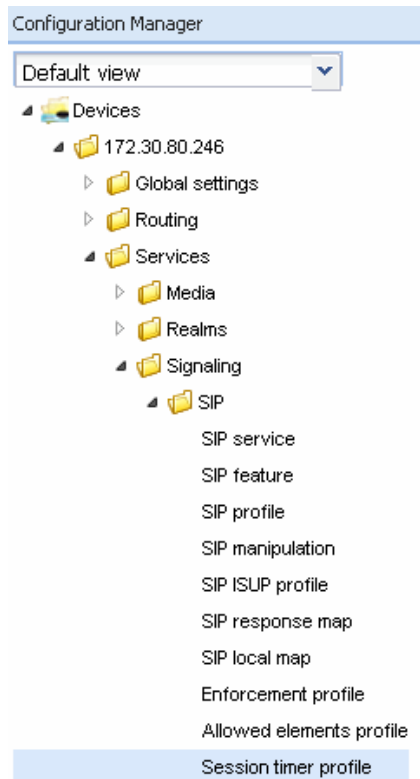
Session Timers Support

Configuring Session Timer Profiles

You configure session timer profiles in SIP signaling. Once configured, you associate a session timer profile to a SIP interface.

To configure session timers support in SIP signaling:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click Session timer profile. The Session timer profile table displays in the content area.
3. Click **Add**. The Add Session timer profile dialog box appears.

4. **Name**—Enter a unique name for this session timer profile. You can use alphabetical or alphanumeric characters.

*Name

bedford73

5. Click **Apply**. A Success message appears.
6. Click **OK**. The session timer profile is added to the Session timer profile table.
7. Click the session timer profile you want to configure and click **Edit**.
8. **Session duration (sec)**—Enter the session duration you want to set for this session timer profile. The valid range is 90 seconds through 999999999 seconds. The default value is 1800 seconds.
9. **Minimum session duration (sec)**—Enter the minimum session duration for this session timer profile. The valid range is 64 seconds through 999999999 seconds. The default value is 90 seconds.
10. **Force re-INVITE**—Click **enabled** or **disabled** in the drop down list to enable or disable **Force re-INVITE** for this session timer profile. The default value is **disabled**.
11. **Request refresher**—Click **uac**, **uas**, or **none** in the drop down list. The default value is **uac**.
12. **Response refresher**—Click **uac**, **uas**, or **none** in the drop down list. The default value is **uas**.

Session duration (sec)

1800

Minimum session duration (sec)

90

Force re-INVITE

disabled

Request refresher

uac

Response refresher

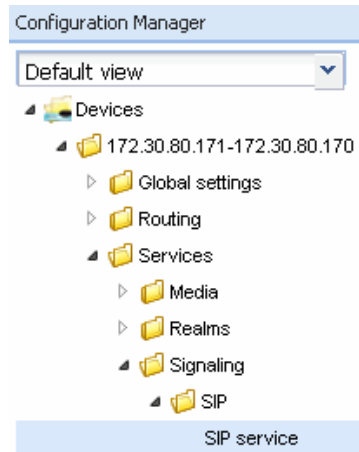
uas

13. Click **Apply**. A Success message appears.
14. Click **OK**.

Associating Session Timer Profiles

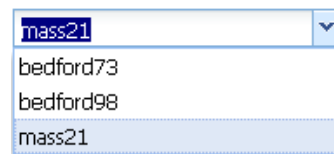
To associate session timer profiles to a SIP interface:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP Interface tab.
4. Click the SIP interface you want to configure in the SIP Interface table and click **Edit**.
5. Click the General tab.
6. Scroll to the bottom of the page.
7. **Session timer profile**—Click the session timer profile in the drop down list that you want to associate with this SIP interface.

Session timer profile



8. Click **Apply**. A Success message appears.
9. Click **OK**.


```
REGISTER sip:512@172.16.101.23:5060 SIP/2.0
Via: SIP/2.0/UDP 172.16.101.38:5070;branch=dd1;keep
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512" <sip:512@172.16.101.38:5070>
Call-ID: 1-14400@172.16.101.38
CSeq: 1 REGISTER
Max-Forwards: 70
User-Agent: ADTRAN_Total_Access_908e_(2nd_Gen)/A1.02.00.E
Content-Length: 0
```

The SBC forwards the Registration request (absent the *keep* parameter) to the Registrar.

```
REGISTER sip:512@192.168.7.32:5060 SIP/2.0
Via: SIP/2.0/UDP
192.168.101.23:5060;branch=z9hG4bK3q43klh3dafekdepnbaqip04e1
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512" <sip:512@172.16.101.38:5070>
Call-ID: 1-14400@172.16.101.38
CSeq: 1 REGISTER
Max-Forwards: 69
Contact: "512"
<sip:512@192.168.101.23:5060;transport=udp>;expires:18000
User-Agent: ADTRAN_Total_Access_908e_(2nd_Gen)/A1.02.00.E
Content-Length: 0
Allow: ACK, BYE, CANCEL, INFO, INVITE, NOTIFY, OPTIONS, REFER
```

The Registrar indicates successful registration with a 200 OK response back to the SBC. The *expires* parameter in the *Contact* header grants a registration period of 1 hour (3600 seconds).

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP
192.168.101.23:5060;branch=z9hG4bK3q43klh3dafekdepnbaqip04e1
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512"
<sip:512@172.16.101.38:5070>;tag=b68b3d53a5a90225609112ff6c211bef.16a6
Call-ID: 1-14400@172.16.101.38
CSeq: 1 REGISTER
Contact: <sip:512@192.168.101.23:5060;transport=udp>;expires=3600
Server: OpenSER (1.3.0-notls (i386/linux))
Content-Length: 0
```

The SBC, forwards the 200 OK to the endpoint after inserting a *keep* parameter and a parameter value in the *Via* header of the Registration response. The presence of the *keep* parameter signals the SBC's willingness to exchange keep-alives, and the parameter value specifies the exchange frequency in seconds.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 172.16.101.38:5070;branch=dd1;keep=20
From: "512" <sip:512@172.16.101.38:5070;transport=UDP>;tag=443322
To: "512"
<sip:512@172.16.101.38:5070>;tag=b68b3d53a5a90225609112ff6c211bef.16a6
Call-ID: 1-14400@172.16.101.38
CSeq: 1 REGISTER
Contact: <sip:512@172.16.101.38:5070>;expires=3600
Server: OpenSER (1.3.0-notls (i386/linux))
Content-Length: 0
```

After the keep-alive exchange has been negotiated, the SIP endpoint, acting as a CR/LF client, is required to transmit a periodic CR/LF so that the interval between each request is randomly distributed between 80 and 100 percent of the value of the *keep* parameter. Assuming a parameter value of 20 seconds, for example, the SIP endpoint transmits a CR/LF at random intervals between 16 and 20 seconds in length.

Upon receipt of a Ping, the SBC, transmits a Pong. Receipt of the Pong by the endpoint confirms the TCP connection between the endpoint and the SBC, and the viability of NAT bindings in the transmission path.

Once initiated, endpoint transmission of CR/LF Ping and SBC responses continue for the duration of the SIP Registration, 1 hour in the above example, or until the endpoint transmits a new Registration request. In the event of such a request, the endpoint must once again indicate its willingness to exchange CR/LF keep-alives with an unvalued *keep* parameter in the *Via* header. If keep-alive renegotiation is not successful, the endpoint must cease the transmission of keep-alive messages.

An endpoint failure to issue a timely CR/LF Ping is not fatal. In the absence of an expected request, the SBC takes no action with regard to the TCP connection, or to established sessions.

Enabling CR/LF on SIP Interfaces

You use the **registration keepalive** parameter, available in Configuration Manager slider -> Services -> Signaling -> SIP, to enable CR/LF keep-alive on a SIP interface.

The **registration keepalive** parameter enables CR/LF keep-alive on the current SIP interface. The valid values are:

none — (the default) disables CR/LF keep-alive

always — assuming that the endpoint has included the *keep* parameter in the *Via* header, exchange CR/LF keep-alives with that endpoint

hint — assuming that the endpoint has included the *keep* parameter in the *Via* header, exchange CR/LF keep-alives only with endpoints that are behind an intervening NAT device (based on comparing source IP packet addresses with IP addresses extracted from the SIP request).

If CR/LF keep-alive is enabled on the current SIP interface (**registration keepalive** is *always* or *hint*), use the **tcp-nat-interval** parameter (from the ACLI view) to specify the value of the *keep* parameter provided by the SBC to the SIP endpoint.

The SIP endpoint transmits periodic CR/LF Ping so that the interval between each request is randomly distributed between 80 and 100 percent of the value of the **tcp-nat-interval** parameter.

To specify the value of the keep parameter provided by the SBC to the SIP endpoint:

1. Click Configuration Manager (ACLI view) > session-router > sip-interface.
2. Scroll to **tcp-nat-interval**.
3. **tcp-nat-interval**—Enter a value (in seconds). Assuming the default value (30 seconds), the interval between CR/LF binding requests would vary from 24 to 30 seconds.

tcp-nat-interval

4. Click **Apply**.

Session Load Balancer

This section contains information about the content of Release L-Cx1.1.3 F2.

As Acme Packet customers deploy larger and larger SIP access networks, scalability problems are presenting unique challenges, particularly from an operational standpoint. Deployments that scale beyond the number of users serviceable by one Acme Packet SBC – as well as deployments that use a geographically redundant Acme Packet SBC for catastrophic fail over purposes – encounter “edge reachability” problems. In general there are two coarse techniques that carriers use today to support endpoint populations that exceed one SBC’s capacity: they will either use a DNS-based distribution mechanism (something that Acme Packet itself has advocated), or they will pre-provision endpoints to point to specific SBCs (manually load balancing them). Each of these solutions has its drawbacks. Acme Packet’s customers – many of them familiar with load balancing equipment deployed to scale protocols such as HTTP or SMTP – have expressed interest in a device that will perform a load balancing function for their SIP endpoints.

Acme Packet developed the Session-aware Load Balancer (SLB) to address the need for scaling a network edge to millions of endpoints. Designed as a standalone system capable of supporting up to two million users (where a “user” is defined as a unique source IP address), and flexibly deployable into existing network topologies, the SLB aggregates signaling from large user populations to reduce the edge reachability problem by an order of magnitude.

Functional Overview

The Session Load Balancer (SLB) is a discrete network element that is the *first logical hop* for all *signaling traffic* from user agents entering a carrier network (and likewise, the *last logical hop* for all signaling traffic exiting the network and destined for user agents). The SLB, upon receipt of a SIP packet from an unknown source, uses a provisioned policy, running on a host processor software component, to select an appropriate next-hop Session Border Controller (SBC) for traffic originated by that source IP address, also referred to as an *endpoint*. Subsequent packets from that endpoint are forwarded to the same SBC without further host involvement. The first packet, the one used to make the route decision, and all subsequent packets sent through the SLB to the next-hop SBC are encapsulated within an IP-in-IP format as defined in RFC 2003, *IP Encapsulation within IP*.

SBCs that participate in the load balancing-enabled deployment are enhanced by several new capabilities. First, the SBC supports the RFC 2003 tunnel for both packet transmission and reception. Second, the SBC periodically transmits health and performance data to the SLB; such information is evaluated and entered into the SLB’s route determination algorithm. Lastly, the SBC participates in an SLB-initiated *rebalance* operation, as described in “Rebalancing” on page 5-792. A group of SBCs, with the above-listed capabilities, that receive signaling traffic from the SLB is referred to as a *cluster*.

The IP-in-IP encapsulation technique provides SLB transparency to the terminating SBC. That is, when an SBC receives an encapsulated packet via the SLB, it can discard the outer encapsulation leaving behind an *identical* packet as transmitted originally by the endpoint. Visibility into the actual packet transmitted by the endpoint is necessary to provide certain services in the SBC (for example, hosted NAT traversal, session-agent matching, and so on). A secondary goal achieved by using this encapsulation technique is that it provides a disassociation function between an SBC’s connected network and its SIP reachability. That is, an SBC can be assigned any IP address it wants from a network topology standpoint, yet still

process SIP packets as though it were logically situated elsewhere at Layer 5. In a larger sense, the *physicality* of the SBC is no longer important; like-configured, logically identical SBCs can be spread all over the globe.

Balancing and Rebalancing

The SLB performs two primary functions as the front-end to an SBC cluster: balancing traffic and rebalancing traffic. There are several key distinctions, which are described in the following two sections.

Balancing

Balancing is defined as the distribution of new endpoints (IP addresses) among the members of the SBC cluster. The SLB balances traffic based upon its configured policies (refer to “Load Balancer Policy Configuration” on page 5-800 for policy description and details), or, in the absence of configured policies, with a default round-robin procedure. Load balancer policies provide a flexible means of directing traffic to appropriate groups of SBCs. As initial packets arrive at the SLB from unknown (previously unseen) endpoints, they are passed to a resident software process that consults its policy engine to determine an appropriate destination (clustered SBC) for each endpoint. Regardless of the distribution algorithm, policy-based or round-robin, the SLB chooses an SBC from among all equally weighted candidates, giving preference to those with the lowest current *occupancy rate*, defined as the number of endpoints already present on that system relative to its maximum number of endpoints.

Even though each clustered SBC regularly reports CPU date to the SLB, the SBC’s CPU utilization is not factored into the preference of one SBC over another. Rather, an SBC whose CPU utilization rate exceeds its load limit threshold (by default, 90%) is excluded from the list of candidates. For example, assuming that both SBCs are licensed for the same number of sessions, an SBC with a CPU load of 89% and a current occupancy of 10,000 endpoints will have equal footing with an SBC with a CPU load of 10% and a current occupancy of 10,000 endpoints. But an SBC with a CPU load of 90% and an occupancy of 0 endpoints will never receive new assignments from the SLB, until its CPU utilization rate falls below the 90% threshold.

Rebalancing

Rebalancing, as opposed to balancing, is taking some number of existing users from functioning SBCs and redistributing these existing users between current cluster members. Rebalancing occurs when a new SBC joins an existing cluster, or when invoked with the Acme Packet Command Line Interface (ACLI). When an SBC is removed from a cluster, whatever the reason, all of its endpoints are invalidated on the SLB and those users are essentially *balanced* when they revisit the SLB.)

A new SBC joins an existing cluster in response to a parameter contained in its start-up configuration that initiates the establishment of an IP-in-IP tunnel between the SBC and the SLB. During an initial handshake the SBC advertises which services it is configured to support. If there are existing SBCs offering this same service, and the SLB is configured to automatically rebalance, the SLB requests each of the SBCs offering this service to divest themselves of a quantity of their subscribers. The SLB chooses the target quantity based upon the overall occupancy of that service relative to the SBC’s contribution to that occupancy. It sends the requests using Cluster Control Protocol messaging to each SBC, instructing each SBC to *nominate* candidates for removal/redistribution. Each SBC in turn replies to the Cluster Control messages with a list of candidates; each Cluster Control reply will have approximately 80 candidates. The SLB removes existing forwarding rules associated

with those users, and repeats this request/response process with each SBC until it has received the target quantity from each. When the affected users re-engage with the SLB (upon their next scheduled registration refresh, for example), the SLB lacks a forwarding rule that maps them to a specific SBC. Consequently, the message is passed up to the software processes running on the SLB's host, which choose a new destination for that user – presumably, the new SBC that has the most open capacity.

The SBC, when requested to nominate users for rebalancing, uses several criteria for choosing the most attractive candidates. As part of its standard session border control behavior the SBC *knows* the expiry times for all of the entries in its registration cache. Hence, the SBC can predict with a high degree of precision when any given user will be signaling back into the cluster. As the forwarding rules on the SLB are instantiated based upon the stimulus of a message from an endpoint, the SBC considers a user whose registration entry is most imminently due to expire a particularly attractive candidate. Note, however, that in many cases it is not prudent to nominate users whose registration cache entries are due to expire *immediately*, as this can cause a race condition between the removal message transmission from the SBC to the SLB and the refresh REGISTER message from the endpoint to the SLB. To avoid this potential dilemma, SBC are equipped with the ability to skip ahead to candidates whose expiry is not immediate.

Further, each SBC categorizes the endpoints stored in its cache based upon a priority value that is determined via the SLB's distribution policy (see "Distribution Policy Configuration" on page 5-802 for more details). It will choose endpoints from its lowest priority buckets first.

Lastly, the SBC does not rebalance a user that is actively in a phone conversation; it would be quite problematic to have in-dialog transactions for an active call arrive at a new SBC that did not handle the original call setup.

SLB Configuration

This section explains how to configure functionality specific to the SLB; it does not include configuration steps for elements that it shares in common with its corresponding SBCs (for example, system configuration, physical interface, network interface, etc.) For information about general SBC configuration, refer to the Acme Packet documentation set as listed in *About This Guide*.

SLB configuration is quite simple; aside from basic network connectivity, the service interfaces, and the distribution policy, much of the configuration is otherwise learned dynamically from the SBCs that comprise the cluster.

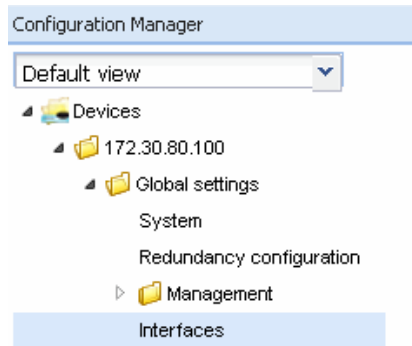
Tunnel Configuration

The SLB sends and receives signaling messages to and from clustered SBCs through an IP-in-IP tunnel. The SLB requires one tunnel per interface per protocol. Currently SIP and H.248 are the only supported protocols. Consequently, running SIP and H.248 simultaneously on one SBC necessitates two parallel tunnels between the SLB and SBC, one tunnel reserved from SIP traffic, and the other for H.248 traffic.

Use the following procedure to perform required SLB-side *tunnel* configuration. Completion of tunnel configuration is accomplished on the clustered SBCs as described in [SBC Tunnel Configuration \(806\)](#).

To configure tunnel configurations:

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.

You must select the physical interface whose network interface you want to configure. (Network interface(s) are associated with a physical interface.)

3. Click the physical interface whose network interfaces you want to configure. The network interfaces associated with this physical interface will appear in the Network interfaces table.
4. Click the network interface you want to configure in the Network interfaces table and click **Edit**.
5. Scroll to the Tunnel config table.

6. Click **Add**. The Add tunnel-config dialog box appears.

7. **Local IP address**—Enter the IP address at the SLB end of the tunnel.

As the terminus for all tunnels from the clustered SBCs, and never the tunnel originator, only the local IP address is configured on the SLB.

*Local IP address

172.30.80.73

8. Click **Apply**. A Success dialog box appears.
9. Click **OK**. The local IP address is added to the Tunnel config table.
10. Click the tunnel configuration you want to configure in the Tunnel config table and click **Edit**.
11. **Port**—Enter the port used to send and receive cluster control messages.

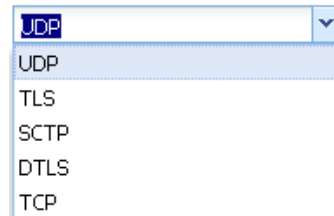
Port

4444

12. **Transport protocol**—Click the transport protocol in the drop down list used in support of cluster control messages.

Supported protocols are UDP (the default), TCP, or TLS.

Transport protocol



Note: Cluster control messages are exchanged quite frequently, and the overhead associated with encrypting and decrypting these messages is significant. Selection of TLS as the Transport Layer protocol degrades system performance.

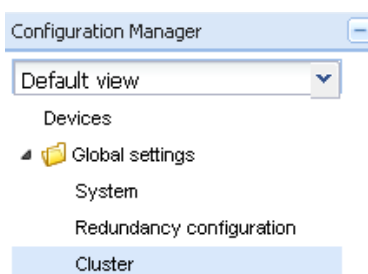
13. **TLS-profile**—If TLS is the selected transport protocol, use this parameter to select the existing TLS profile that identifies the cryptographic resources used to secure the TLS connection.
14. Click **Apply**.
15. Repeat steps 3 through 14 to configure additional tunnel configurations.

Cluster Configuration

The cluster configuration element manages basic SLB interaction with clustered SBCs — it contains a set of global parameters that define the management of the RFC 2003 IP-in-IP tunnels that connect the SLB to clustered SBCs, and the details of rebalance operations. In addition, cluster configuration provides for the creation of a list of service interfaces (signaling addresses) that are advertised to endpoints comprising the user access population.

To configure a cluster configuration:

1. Expand the Configuration Manager slider > Global settings.
2. Click Cluster.



3. **State**—Click enabled or disabled in the drop down list to enable or disable the SLB software.

The default setting, **enabled**, enables SLB functionality; **disabled** renders the SLB inoperable.

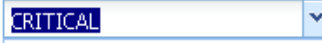
State enabled ▼

4. **Log level**—Click the log level in the drop down list to specify the contents of the SLB log.

Log messages are listed below in descending order of severity.

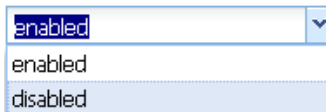
emergency	— the most severe
critical	
major (error)	
minor (error)	
warning	
notice	
info	— (default) the least severe
trace	— (test/debug, not used in production environments)
debug	— (test/debug, not used in production environments)
detail	— (test/debug, not used in production environments)

In the absence of an explicitly configured value, **Log level** defaults to *critical*, meaning that log messages with a severity of *critical* or greater (*emergency*) are written to the SLB log.

Log level 

5. **Auto rebalance**—Click enabled or disabled in the drop down list to specify SLB behavior when a new SBC joins an existing cluster. With this parameter:

- **Enabled**, the default setting: The SLB redistributes endpoints among cluster members when a new member joins the cluster. Refer to “Rebalancing” on page 5-792 for operational details.
- **Disabled**, the alternate setting: Pre-existing SBCs retain their endpoint populations, and the SLB directs all new endpoints to the newly active SBC until that SBC reaches maximum occupancy.

Auto rebalance 

Note: If **auto rebalance** is set to **disabled**, the following three parameters can be ignored.

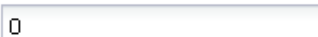
6. **Source rebalance threshold**—If **auto-rebalance** is set to **enabled**, use the **source rebalance threshold** to specify threshold settings that identify existing cluster SBCs as endpoint sources.

The **source rebalance threshold** parameter specifies the minimum occupancy rate that identifies a clustered SBC as a source of endpoints during a rebalance operation. For example, using the default value of 50 (percent), any clustered SBC with an occupancy rate of 50% or more sheds endpoints during a rebalance. The SLB assigns these endpoints to the new cluster member.

Source rebalance threshold 

7. **Destination rebalance threshold**—If **auto-rebalance** is set to **enabled**, use the **destination rebalance threshold** parameter to specify the endpoint destinations during the rebalance operation.

The **destination rebalance threshold** parameter specifies the maximum occupancy rate that identifies a clustered SBC as a destination of endpoints during a rebalance operation. Note that using the default setting of 0 (percent), ensures that no pre-existing SBC gains endpoints during a rebalance.

Destination rebalance threshold 

8. **Destination rebalance max**—If **auto-rebalance** is set to **enabled**, use the **destination rebalance max** parameter to specify the occupancy rate for the new cluster member.

The **destination rebalance max** parameter specifies the maximum occupancy rate that the SLB transfers to the new cluster member during a rebalance operation. The default setting is 80 (percent). Should this threshold value be attained, the SLB distributes remaining endpoints to those SBCs identified as endpoint destinations by their **destination rebalance threshold** settings. Refer to [Balancing and Rebalancing \(792\)](#) for details on occupancy and its calculation.

Parameter values are numeric percentages within the range 0 through 100.

Destination rebalance max 

Note: If **auto-rebalance** is set to **disabled**, the following four parameters can be ignored. If **auto-rebalance** is set to **enabled**, you can optionally use the following four parameters to fine-tune rebalance operational details.

9. **Rebalance request delay (ms)**—Enter the interval (in milliseconds) between endpoint request messages sent from the SLB to a clustered SBC. As explained in [Rebalancing \(792\)](#), these messages request a list of endpoints that will be redistributed from the SBC to a new cluster member. By default, this parameter is set to 500 milliseconds.

Rebalance request delay (ms)

500

10. **Rebalance skip ahead (ms)**—Enter the interval (in milliseconds) that restricts the target set of SBC endpoints registration eligible for rebalancing to those whose re-registration is not imminent, that is, the registration is not scheduled within the number of milliseconds specified by the parameter setting. Setting this parameter to a non-zero value mitigates against the possibility of a race condition precipitated by a simultaneous endpoint removal generated by the SBC and the arrival of endpoint signalling on an SLB service port. The default setting (0 milliseconds) effectively makes the entire SBC endpoint set eligible for rebalancing.

Rebalance skip ahead (ms)

0

11. **Rebalance maximum refresh (ms)**—Enter the interval (in milliseconds) that restricts the target set of SBC endpoints eligible for rebalancing to those whose re-registration is far in the future, that is, the registration is not scheduled until after the number of milliseconds specified by the parameter setting. The default setting (0 milliseconds) effectively makes the entire SBC endpoint set eligible for rebalancing.

Rebalance maximum refresh (ms)

0

12. **Delete application entries while rebalance**—Click enabled or disabled in the drop down list to specify when cached SIP entries for rebalanced endpoints are removed from the clustered SBC. The default setting (**disabled**) specifies that cached entries are retained after a rebalance operation, and subsequently removed from the cache by standard time-out procedures. When set to **enabled**, this parameter specifies that the SBC removes cached registration entries at the completion of the rebalance operation.

Delete application entries while rebalance

disabled



Three parameters, **tunnel fail interval (ms)**, **tunnel check interval (ms)**, and **inactive SD limit (ms)** maintain and monitor the IP-in-IP tunnels established between the SLB and clustered SBCs.

13. **Tunnel fail interval (ms)**—Enter the interval (in milliseconds) between periodic *keepalive* messages sent from a clustered SBC to the SLB. If the SLB fails to receive a *keepalive* message within the specified period, it flags the tunnel as *dead*. By default, this parameter is set to **10000** milliseconds.

Tunnel fail interval (ms)

10000

14. **Tunnel check interval (ms)**—Enter the interval (in milliseconds) between SLB tunnel audits. During a tunnel audit, the SLB checks the status of each tunnel and removes all tunnels flagged as *dead*. By default, this parameter is set to **15000** milliseconds.

If you change default settings for either parameter, ensure that the setting for **tunnel check interval (ms)** is greater than the **tunnel-fail-interval** setting.

Tunnel check interval (ms)

15. **Inactive SD limit (ms)**—Enter the maximum silent interval (in seconds) before the SLB flags the SBC as *dead*, and removes it from the cluster. By default, this parameter is set to 1800 seconds (30 minutes). supported values are integers within the range 0 through 31556926 (365 days).

Inactive SD limit (ms)

The **session multiplier** and **atom limit divisor** parameters specify a numeric factors used in *occupancy* and *occupancy rate* calculations.

16. **Session multiplier**—Enter a factor that when multiplied by an SBC's licensed session limit, determines the maximum number of endpoints that the SBC can support (that is, its maximum *occupancy*).

The default setting is 10; valid settings include any integer values within the range 1 through 100.

Using the default setting, an SBC licensed for 32,000 concurrent sessions has a maximum theoretical occupancy of 320,000 endpoints.

Session multiplier

17. **Atom limit divisor**—Enter a factor that can be used in *occupancy* and *occupancy rate* calculations. By default, occupancy calculations are based on endpoints (IP addresses), and do not take into account the fact that the same IP address can represent multiple users.

The default setting is 1, which assumes a conservative 1-to-1 correlation between endpoints and users; valid settings include any integer values within the range 1 through 1000.

Atom limit divisor

Note: Once an SBC has reached its maximum number of endpoints, the SLB removes it from the load balancing algorithm. These parameter settings should be changed only after careful examination of network conditions and behavior.

Note: The **Ignore target services on rebalance** parameter is not currently supported.

18. Retain default settings for the **Redundancy sync port**, **Max redundant transactions**, and **Redundant sync start timeout**.

Redundant sync port

Max redundant transactions

Redundant sync start timeout

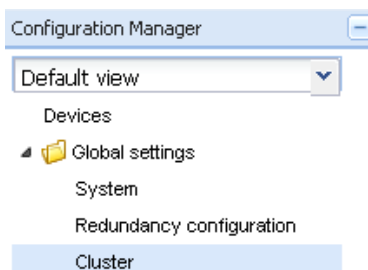
19. Click **Apply** complete cluster configuration.

Service Ports Configuration

A service port is an access-side network interface monitored by the SLB for incoming signaling from the user population. For virtually all network topologies, multiple service ports are expected on a typical SLB configuration. A service port is a multiple instance configuration element; for each service port advertised to the access network(s), at least one service port configuration element must be configured.

To configure service ports:

1. Expand the Configuration Manager slider > Global settings.
2. Click Cluster.



3. Scroll to Service port.
4. Click **Add**. The Add service port dialog box appears.
5. **Address**—Enter the IP address of this service port.
6. **Port**—Enter the port monitored by the SLB for incoming signaling messages.

In the absence of an explicitly configured port, the SLB provides a default value of 5060 (the registered SIP port).

Allowable values are integers within the range 0 through 65535.

7. **Protocol**—Click the transport protocol in the drop down list. The valid values are *UDP* (the default) and *TCP*.

*Address	172.30.82.73
*Port	0
*Protocol	UDP

8. Click **Apply**. A Success dialog box appears.
9. Click **OK**.
10. To configure the network interface, click the service port you want to configure in the Service port table and click **Edit**.
11. **Network interface**—Enter the SLB network interface that supports this service port.

Network interface	:0

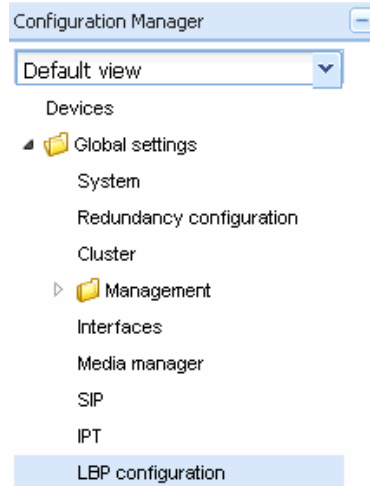
12. Click **Apply**.
13. Repeat steps 4 through 12 to configure additional service port configurations.

Load Balancer Policy Configuration

The LBP configuration element manages the SLB endpoint table. It also creates and manages a list of service interfaces (signaling addresses) that are advertised to endpoints comprising the user access population.

To configure an LBP configuration:

1. Expand the Configuration Manager slider > Global settings.
2. Click LBP configuration



3. **State**—Click enabled or disabled to enable or disable the LBP software.
The default setting, **enabled**, enables SLB functionality; **disabled** renders the SLB inoperable.
4. **Log level**—Click the log level in the drop down list to specify the contents of the SLB log.

Log messages are listed below in descending order of severity.

emergency	— the most severe
critical	
major (error)	
minor (error)	
warning	
notice	
info	— (default) the least severe
trace	— (test/debug, not used in production environments)
debug	— (test/debug, not used in production environments)
detail	— (test/debug, not used in production environments)

In the absence of an explicitly configured value, **log level** defaults to **critical**, meaning that log messages with a severity of **critical** or greater (**emergency**) are written to the LBP log.

Log level CRITICAL

The **untrusted grace period (sec)**, **maximum untrusted percentage**, **maximum untrusted upper threshold**, and **maximum untrusted lower-threshold** parameters implement percentage-based management and monitoring of untrusted endpoints in the SLB endpoint database. Management and monitoring of untrusted endpoints is instrumental in detecting and responding to Denial-of-Service (DOS) attacks aimed at the SLB.

5. **Untrusted grace period (sec)**—Enter the maximum time, in seconds, that a forwarding rule is retained by the SLB before it is confirmed with a promotion message from the SBC which received the untrusted endpoint. Refer to [Balancing \(792\)](#) for message details.

In the absence of an explicitly assigned value, the SLB provides a default setting of **30** (seconds).

If this time period elapses without a promotion message arriving to confirm this user, the SLB deletes the entry.

Setting this parameter to 0 allows untrusted/unconfirmed entries to exist indefinitely without aging out.

Untrusted grace period (sec)

6. **Maximum untrusted percentage**—Enter the percentage of the overall endpoint population that is reserved for untrusted users.

The default setting is **20** (percent); supported values are integers within the range 1 through 100.

This percentage is applied to the overall remaining occupancy of the SLB after trusted (confirmed) users are accounted for. For example, when empty, the SLB holds two million forwarding rules; assuming the default setting, at most 400,000 rules are reserved for untrusted rules. By the time one million users have been promoted, 20% of the remaining space means that up to 200,000 entries can be used for untrusted users.

Maximum untrusted percentage

7. **Maximum untrusted upper threshold**—Enter a threshold level at which the SLB (1) raises an alarm, and (2) issues an SNMP trap reporting an excessive number of untrusted endpoints within the entire endpoint population.

This parameter, which has a default setting of **80** (percent), is calculated as a percent of **maximum untrusted percentage**. For example, assuming default settings for both parameters, the SLB raises an alarm and issues an SNMP trap when the percentage of untrusted endpoints attains 16%.

Maximum untrusted upper threshold

8. **Maximum untrusted lower threshold**—Enter a threshold level at which the SLB (1) clears the existing untrusted endpoint alarm, and (2) issues an SNMP trap reporting alarm clearance.

This parameter, which has a default setting of **70** (percent), is calculated as a percent of **maximum untrusted percentage**. For example, assuming default settings for both parameters, the SLB clears an alarm and issues an SNMP trap when the percentage of untrusted endpoints falls to 14%.

Maximum untrusted lower threshold

The **endpoint capacity upper threshold** and **endpoint capacity lower threshold** implement license-based management and monitoring of the SLB endpoint counts.

9. **Endpoint capacity upper threshold**—Enter a threshold level at which the SLB (1) raises an alarm, and (2) issues an SNMP trap reporting an excessive number of active endpoints.

This parameter, which has a default setting of **80** (percent), is calculated as a percentage of the endpoints allowed by the installed SLB license.

Endpoint capacity upper threshold

10. **Endpoint capacity lower threshold**— Enter a threshold level at which the SLB (1) clears the existing endpoint alarm, and (2) issues an SNMP trap reporting alarm clearance.

This parameter, which has a default setting of **70** (percent), is calculated as a percentage of the endpoints allowed by the installed SLB license.

Endpoint capacity lower threshold

11. Click **Apply** to complete configuration of this load balancer policy configuration.

Distribution Policy Configuration

Distributing endpoints equitably among the cluster members is the primary function of the SLB. The LP configuration element allows you to control the method of the SLB's distribution based on matching criteria. Using inbound packet matching criteria, you can control the assignment of users to SBCs. Matching is done by data available up to and including the transport layer of the packet: source IP address and port, destination IP address and port, and transport protocol. The IP addresses and ports may or may not include bit masks as well.

Conceptually, the load balancer policy table, with sample data, looks akin to the following.

Source IP/Mask	Source Port/Mask	Destination IP/Mask	Destination Port/Mask	Transport Protocol Requirements (list)	Realm Identifiers (list)
192.168.7.22/32	0/0	10.0.0.1/32	5060/16		West
192.168.1.0/24	0/0	10.0.0.1/32	5060/16	UDP, TCP	North, South, West
192.168.0.0/16	0/0	10.0.0.1/32	5060/16	UDP, TCP	East, West
0.0.0.0/0	0/0	0.0.0.0/0	0/0		

Policies are matched using a longest prefix match algorithm; the most specific policy is selected when comparing policies to received packets. One and only one policy is chosen per packet; if the next hops in that route are all unavailable, the next best route is not consulted (instead, the *default policy* may be consulted – see below). This is different than the local-policy behavior on the SBC.

Within each policy you may configure multiple next hops, where each next hop is a named group of SBCs. In the sample policy table, this is indicated in the second policy with a source IP range of 192.168.1.0/24. The realm identifier list for this policy indicates North, South, West. Each of these realm identifiers represents a collection of zero or more SBCs, in SBC parlance these are roughly analogous to *session-agent groups*. Each of these realm identifiers is also assigned a *priority* (a value between 1 and 31, with 1 representing the highest priority) in the configuration, and the SLB sorts the possible destinations with the highest priority first. Upon receipt of a packet matching a policy with multiple configured realm identifiers, the SLB gives preference to SBCs from the realm identifier with the highest priority. Should no SBCs be available in that priority level (due to saturation, unavailability, and so on.) the SLB moves on to investigate the next priority level, and so on. Should no SBCs

become available after traversing the entire list of all SBCs within each priority level, the SBC either drops the packet or attempt to use the *default policy*.

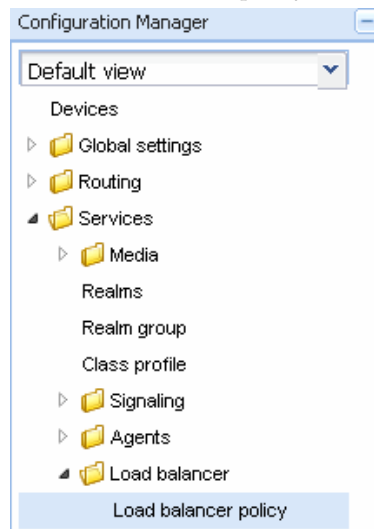
The bottom row of the sample table shows this implicit, last resort default policy. When enabled, the SLB reverts to the default policy when all of the potential next hop realms referenced in the endpoint's distribution rule are unavailable. In that event, the default policy attempts to locate a clustered SBC that advertises support for the service-interface that the packet arrived on. The realm is not considered when matching to the default policy. If such an SBC is found, the SLB forwards the packet to that SBC; if such an SBC is not found, the SLB drops the packet.

It is not necessary to configure the default policy — it is simply intended as a catchall policy, and may be used when all that is required is a simple round-robin balancing scheme based on simple metrics (for example, CPU utilization and number of registrations currently hosted by an SBC). If no policies are configured on the SLB, the default policy is used. The default realm is implied in the above table as "*" and is enabled by default for policy records.

Adding a Load Balancer Policy

To add a load balancer policy:

1. Expand the Configuration Manager > Services > Load balancer.
2. Click Load balancer policy.



3. Click **Add**. The Add lb-policy dialog box appears.
First, you will configure the Source parameters and then the Destination parameters.
4. **port**—Enter the source port for this load balancer policy. The valid range is 0 through 65535. The default value is 0.
5. **address**—Enter a valid IP address for this load balancer policy. The default value is 0.0.0.0.
6. **number of bits**—Enter the number of bits for this load balancer policy. The valid range is 0 to 32 bits. The default value is 32.
Perform the following three steps to configure the Destination parameters.
7. **port**—Enter the destination port for this load balancer policy. The valid range is 0 through 65535. The default value is 0.

8. **address**—Enter a valid IP address for this load balancer policy. The default value is 0.0.0.0.
9. **number of bits**—Enter the number of bits for this load balancer policy. The valid range is 0 to 32 bits. The default value is 32..

Source

*port	0
*address	0.0.0.0
*numberOfBits	32

Destination

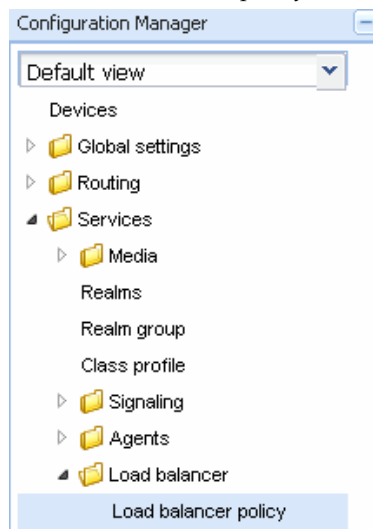
*port	0
*address	0.0.0.0
*numberOfBits	32

10. Click **Apply**. A Success message appears.
11. Click **OK**. The load balancer policy is added to the Load balancer policy table.
12. Repeat steps 3 through 11 to add additional load balancer policies.

Configuring a Load Balancer Policy

To configure a load balancer policy:

1. Expand the Configuration Manager > Services > Load balancer.
2. Click Load balancer policy.



3. Click the load balancer policy you want to configure in the Load balancer policy table and click **Edit**.
4. **State**—Click **enabled** or **disabled** to enable or disable this distribution rule. The default setting, **enabled**, enables the distribution rule; **disabled** disables the rule.

5. **Default realm**—Click **enabled** or **disabled** to enable or disable the default distribution policy.

The default setting, **enabled**, enables the default policy; **disabled** disables the policy.

With **default realm** enabled, the SLB provides a best-effort delivery model if the next-hop realms listed in this distribution rule are unavailable. With **default realm** disabled, the orphaned packet is dropped.

6. **Description (Optional)**—Enter a description of this distribution rule.

State	enabled ▼
Description	description of distribution rule
Default realm	enabled ▼

7. **Protocol**—Click **Add**. The Add protocol dialog box appears.

7a. Click the protocol in the drop down list that must be supported by this distribution rule. The valid values are UDP, SCTP, and TCP.

7b. Click **Apply**. A Success message appears.

7c. Click **OK**. The protocol is added to the Protocols list.

*Protocol	UDP ▼
	UDP
	SCTP
	TCP

8. Repeat steps 7 through 7c to add additional protocols.

9. **Source**—Click the source IP address you want to modify in the Source table and click **Edit**.

9a. **Address**—Enter a valid IP address to specify source-address-based matching criteria.

Packets whose source IP addresses match the criteria specified by this parameter are subject to this distribution rule.

- Matches any port on the specified IP source address
- Matches the specified IP source address:port pair
- Matches any IP source address, any port on the 10.0.0.x subnet
- Matches IP source addresses 10.0.0.240:5060 through 10.0.0.255:5060

10. Click **Apply**. A Success message appears.

11. Click **OK**.

12. **Destination**—Click the destination IP address you want to modify in the Source table and click **Edit**.

12a. **Address**—Enter a valid IP address to specify destination-address-based matching criteria.

Packets whose destination IP addresses match the criteria specified by this parameter are subject to this distribution rule.

- Matches any port on the specified IP destination address

- Matches the specified IP destination address:port pair
- Matches any IP destination address, any port on the 10.0.0.x subnet
- Matches destination IP addresses 10.0.0.240:5060 through 10.0.0.255:5060

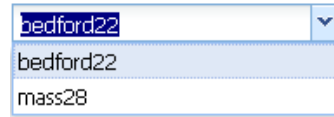
13. Click **Apply**. A Success message appears.

14. Click **OK**.

15. **Realms**—Click **Add**. The Add Realms dialog box appears.

16. **Name**—Click the realm name in the drop down list eligible to receive traffic (that matches this distribution rule).

*Name



The **name** field is roughly analogous to an SBC session-agent group. SBCs configured to communicate within a cluster hosted by an SLB advertise offered services to the SLB. These services (for example, SIP support or H.248 support) exist in realms, whose names are sent to the SLB as part of the SBC advertisement. The SLB, upon receipt of these advertisements, joins each SBC into one or more realm identifier groups based upon the realm name(s) the SBC has offered up. The **name** command of the load balancer policy configuration matches this distribution rule to a supporting SBC that has offered that realm name for cluster membership.

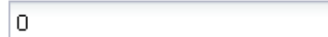
17. **Priority**—Click the realm you want to set priority for and click **Edit**.

17a. **Priority**—Enter an integer value within the range 0 to 3, where the lower the integer, the greater the priority.

The default value, 0, specifies use of the default routing policy, and should not be used when policy-based distribution is enabled.

Priority values are considered when multiple SBCs offer the same service to matched packets.

Priority



18. Click **Apply**.

SBC Configuration

This section describes the configuration necessary to allow an SBC to join a cluster. Configuration is simplified to allow for an easy and seamless migration from a deployed standalone SBC to a deployed clustered SBC. There are only two places where new configuration is required: in the *network-interface* configuration element, where tunnel information is defined; and in the signaling application's interface, either the *sip-interface* configuration element for clustering the SIP protocol or the *h248-mgc-config* configuration element for clustering the H.248 protocol.

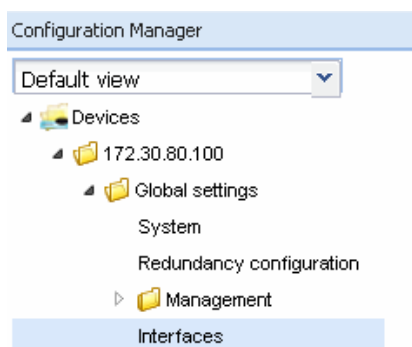
SBC Tunnel Configuration

Configuring the properties of the IP-in-IP tunnel on the SBC is a matter of configuring the local address and specifying transport layer and application layer protocol support.

The following example uses a tunnel named *sipSignaling*, which was initially and partially configured on the SLB. Note in the following configuration that the **Remote IP address** parameter is not specified — that value having been previously set with the **Local IP address** parameter on the SLB. The complementary configuration performed on the SBC enables tunnel establishment between the SBC and the SLB.

To configure tunnel configurations:

1. Expand the Configuration Manager slider > Global settings.



2. Click Interfaces. The Network and Physical interfaces tables appear in the content area.

You must select the physical interface whose network interface you want to configure. (Network interface(s) are associated with a physical interface.)

3. Click the physical interface whose network interfaces you want to configure. The network interfaces associated with this physical interface will appear in the Network interfaces table.
4. Click the network interface you want to configure tunnel configuration for and click **Edit**.
5. Scroll to the Tunnel config table.
6. Click **Add**. The Add tunnel-config dialog box appears.
7. **Local IP address**—Enter the IP address at the SBC end of the tunnel.
8. Click **Apply**. A Success message appears.
9. Click **OK**. The local IP address is added to the tunnel config table.
10. Click the local IP address you want to configure a port for and click **Edit**.
11. **Port**—Enter the port used to send and receive cluster control messages. The default value is 4444.

Port	4444

12. Click **Apply**. A Success message appears.
13. Click **OK**.
14. If TLS is the selected transport protocol, use the **TLS profile** parameter to select the existing TLS profile that identifies the cryptographic resources used to secure the TLS connection.

Note: TLS usage is not recommended!

15. Click **Apply**.

Introduction

The Net-Net SBC supports H.323 signaling in a way that permits interworking between different H.323 configurations from different providers and carriers. H.323 signaling capabilities on the Net-Net SBC include:

- H.323 V4—Improves on previous versions of the protocol in functionality, scalability, and reliability
- H.225 call signaling with RAS—Establishes connections between H.323 endpoints so real-time data can be exchanged
- H.245—Establishes the type of media flow and manages that flow after it has started
- H.245 tunneling—Encapsulates H.245 messages within H.225/Q.931 messages; when enabled and used with a firewall, one less TCP port is needed for incoming connections
- Fast Start (and Fast Start with parallel H.245)
- H.323 Annex E support for UDP signaling—Provides for multiplexed call signaling over UDP to increase potential call volume and enhance performance

Using H.323 on your Net-Net SBC, you can implement different signaling modes and use features to enhance H.323 capabilities. In the information that follows, you will find detailed explanations of the H.323 signaling mode and of the features available. This chapter gives operational details and later outlines the steps you need to take when features require configuration. Certain H.323 features do not require you to set specific parameters; they are described in this chapter's H.323 Automatic Features.

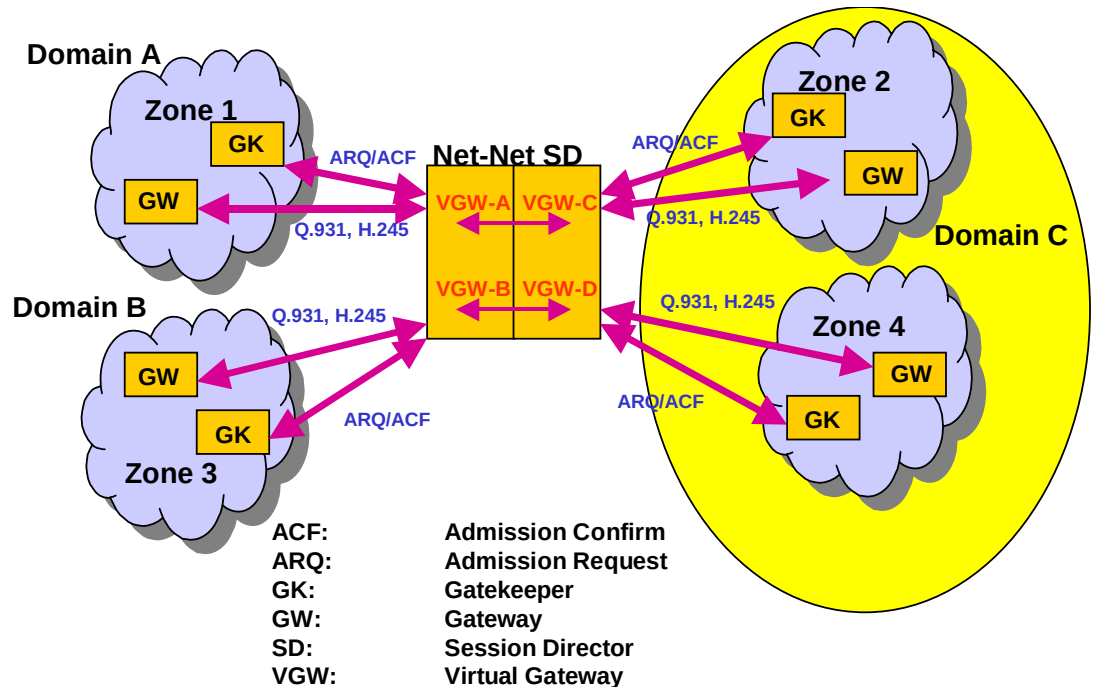
Signaling Modes of Operation

Your Net-Net SBC can operate in different H.323 signaling modes:

- Back-to-back gateway signaling
- Back-to-back gatekeeper proxy and gateway
- Interworking gatekeeper/gateway

Back-to-Back Gateway Signaling

This section explains how signaling takes place when the Net-Net SBC functions as a B2BGW for H.323. The following diagram illustrates the Net-Net SBC acting as a B2BGW.



When configured as a B2BGW, the Net-Net SBC appears as multiple H.323 gateways to multiple networks. You can think of the Net-Net SBC as having virtual gateways, that discovers and registers with a gatekeeper in its respective domain. In this configuration, you need to set the service mode (**isgateway**) parameter for the H.323 interface to enabled for two H.323 interfaces. These interfaces are related either through their outgoing interface (**assoc-stack**) parameters or through routing policies.

If you configure your Net-Net SBC to operate in this mode, it does not issue or respond to LRQs by either confirming them or rejecting them.

In the diagram above, the Net-Net SBC sends ARQs to the corresponding gatekeeper in its zone when a call is received on the associated interface. In this behavior, the Net-Net SBC acts as a gateway, complying with the H.323 standard, and registers with the configured gatekeeper in its assigned zone. You set all parameters related to the gateway registrations, such as gateway prefix numbers, in the H.323 interface configuration.

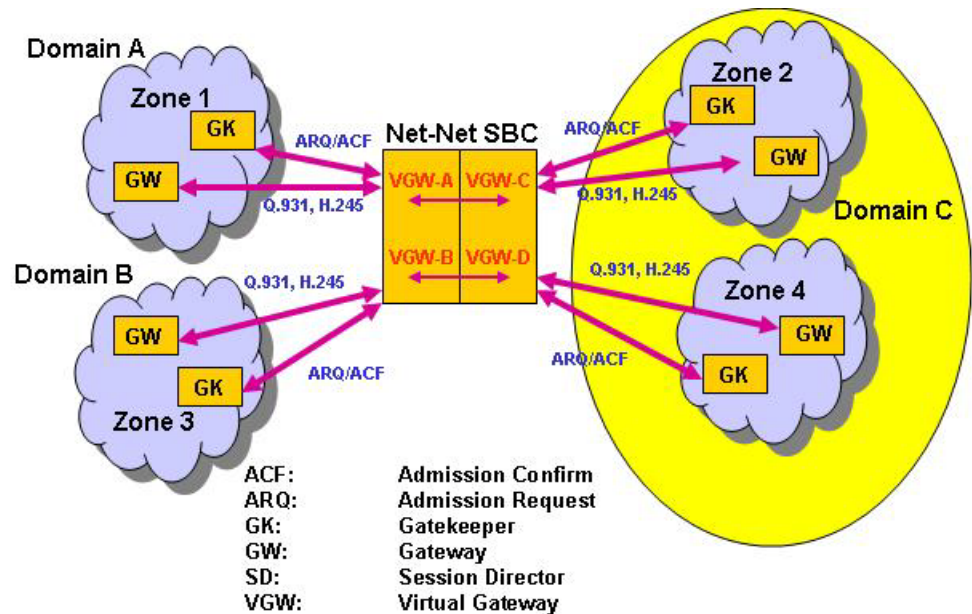
In this mode, you can also configure the Net-Net SBC to run like a gateway without a gatekeeper by turning off automatic discovery (**auto-gk-discovery**) for the remote

gatekeeper. When the Net-Net SBC receives a Setup message, it does not send an ARQ and there is no registration for admission requests. Without automatic gateway discovery, the Net-Net SBC uses the local policy to find the appropriate destination for the call. This destination is normally the IP address of the endpoint or gateway, using the well-known port 1720.

If you enable this capability, then the Net-Net SBC finds a gatekeeper.

Back-to-Back Gatekeeper Proxy and Gateway

This section explains how signaling takes place when the Net-Net SBC functions as a back-to-back gatekeeper proxy and gateway for H.323. The following diagram illustrates the Net-Net SBC acting as a B2B gatekeeper proxy and gateway.



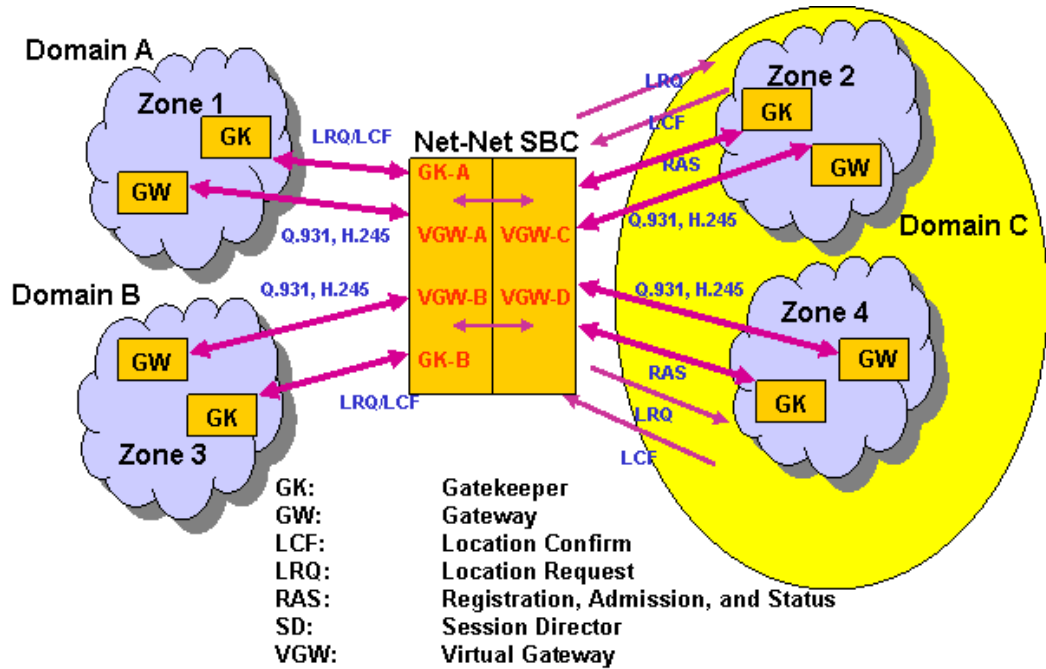
In this application, with the service mode (**isgateway**) parameter set to **disabled**, the Net-Net SBC responds to LRQs and issues LCFs and LRJs. It sends LRQs and LCFs/LRJs to the local IP address for the H.323 interface. The Net-Net SBC responds to the LRQs by providing a signaling address that performs gateway functions.

When you use it as a back-to-back gatekeeper proxy and gateway, the Net-Net SBC does not issue ARQs. In addition, all parameters related to registration, such as gateway prefix numbers, are ignored.

When you do not configure a gatekeeper, the Net-Net SBC uses the local policy to find the appropriate destination for the call. If there is a matching local policy, the Net-Net SBC returns an LCF to the originating gateway. If no local policy matches, the Net-Net SBC rejects the call by sending an LRJ.

Interworking Gatekeeper-Gateway

This section explains how signaling takes place when the Net-Net SBC functions as an interworking gatekeeper-gateway for H.323. The following diagram shows the Net-Net SBC acting as an interworking gatekeeper-gateway.



When you configure your Net-Net SBC for interworking gatekeeper-gateway mode, one H.323 interface behaves as a B2BGW and its associated interface for the corresponding network behaves like a gatekeeper proxy and gateway. The interface for the gatekeeper proxy and gateway issues and responds to LRQ messages on its network. If the Net-Net SBC knows the gatekeeper in the network of the gateway interface (Zone 2), it sends an LRQ to that gatekeeper. If the gatekeeper responds with an LCF or LRJ, the Net-Net SBC forwards it.

If the gatekeeper (in Zone 2) is unknown, then the Net-Net SBC responds to LRQs on the gatekeeper-gateway network (Zone 1) by using the local policy to determine the appropriate destination for the LRQ. If there is no local policy that matches, then the Net-Net SBC sends an LRJ.

For this configuration, the gateway interface has its service mode (**isgateway**) set to **enabled**, and the gatekeeper interface has its service mode (**isgateway**) set to **disabled**.

Before You Configure H.323

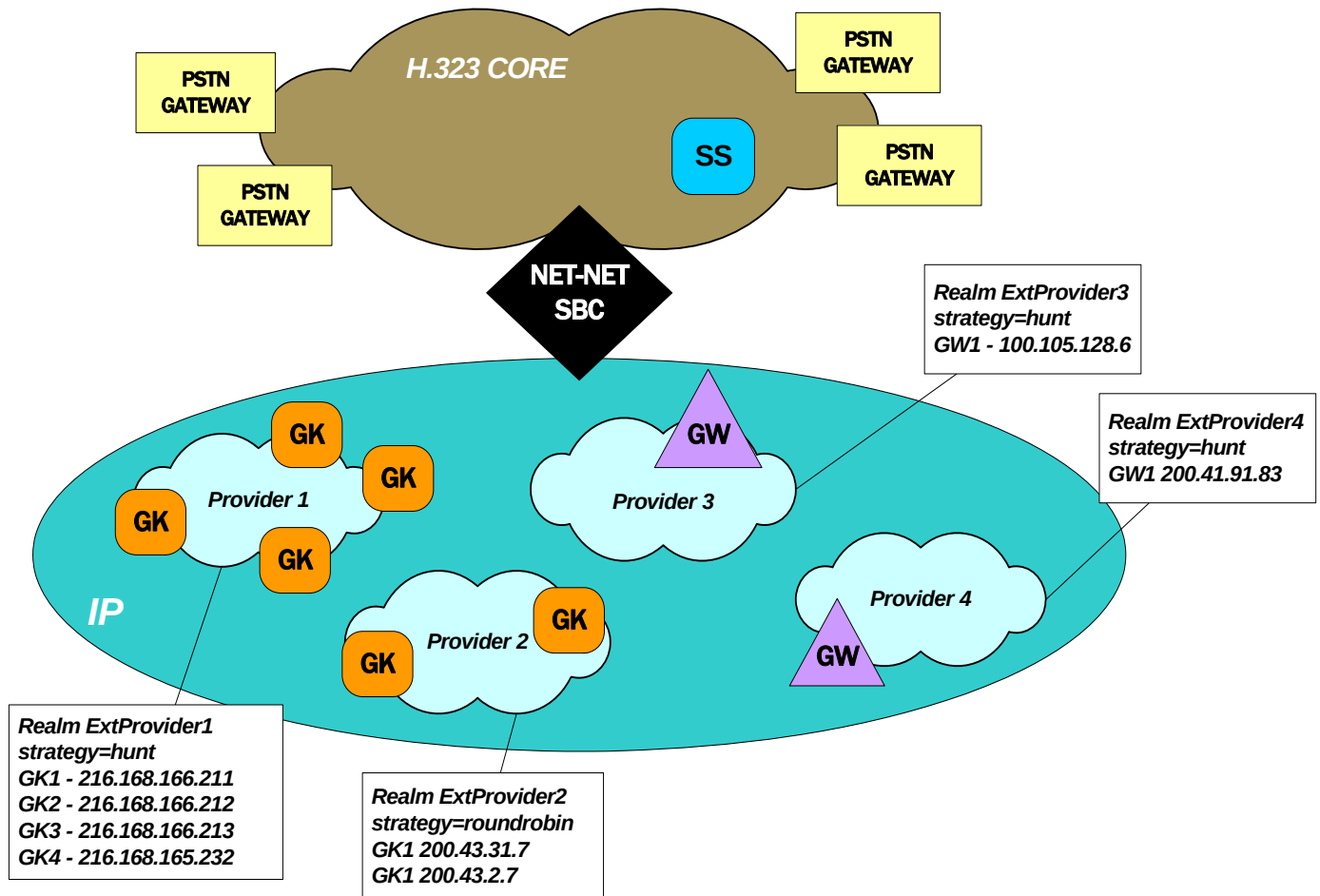
In order to run H.323 on your Net-Net SBC, you need to configure the basic parameters: physical and network interfaces; global system parameters; SNMP, trap receiver, and accounting support, and any holiday information you might want to set.

You should also decide how you want to set up realms and routing (including the use of session agents and session agent groups) to support H.323 operations. For more information about configuring these, refer to the following chapters in this guide:

- Realms and Nested Realms
- Session Routing and Load Balancing

Peering Environment for H.323

The following diagram shows a peering environment for H.323, with the Net-Net SBC positioned between the H.323 core and external providers.



The configuration information shown in the diagram can help you to understand how some basic Net-Net concepts work. The providers in this depiction are configured as realms, and the strategies you see are for session agent group. What you do not see in this diagram is the fact that the Net-Net SBC is configured with

sets of H.323 interfaces within it. These interfaces are internal (for an internal provider) and external (for the external providers you see).

In this chapter's Signaling Modes of Operation, you will learn how the Net-Net SBC can operate the different modes that support this solution.

Realm Bridging with Static and Dynamic Routing

The Net-Net SBC uses static routing and policy-based, dynamic routing to handle H.323 traffic. These types of routing have to do with the way that the outgoing stack is selected.

- Static routing—The incoming H.323 stack always uses the associated H.323 stack that you configure for outgoing traffic; no other stacks are considered.
- Dynamic routing—When there is not an associated stack configured, the Net-Net SBC performs policy-based, dynamic routing known as realm bridging. In this type of realm bridging, the Net-Net SBC checks the configured local policies for address information corresponding to the incoming traffic and finds an address that matches. Next, it checks the next hop in the local policy to determine a realm and uses the first H.323 interface that matches it.

For more information about H.323 and routing, including configuration, refer to this guide's Session Routing and Load Balancing chapter.

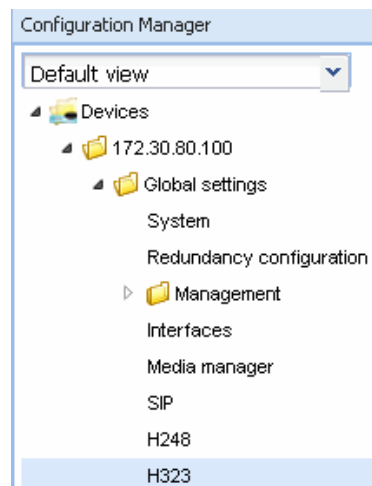
Configuring Global H.323 Parameters

When you configure H.323 signaling for your Net-Net SBC, you set global and per-interface parameters. The global parameters govern how the Net-Net SBC carries out general H.323 operations, and these settings are applied to all interfaces you configure for H.323 use. For example, you can turn H.323 support on and off for the entire Net-Net SBC using these settings.

Configuring Global H.323 Parameters

To configure global H.323 parameters:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.

4. **State**—Select **enabled** from the combination box.
5. **Response to SETUP message sent**—Enter the time in seconds that the Net-Net SBC waits between sending a Setup message and tearing it down after no response.
6. **Call establishment - SETP to CONNECT message interval**—Enter the time in seconds that the Net-Net SBC waits between sending a Setup message and tearing it down if it does not receive a Connect message from the endpoint.
7. **Process log level**—Select the log level for monitoring H.323 activity on the Net-Net SBC from the combination box.

State	enabled
Timeouts(sec)	
Response to SETUP message sent	4
Call establishment-SETUP to CONNECT message interval	32
Process log level	NOTICE

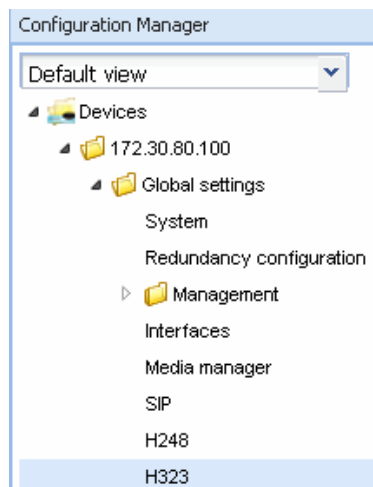
Configuring Global Options

A global option has a global impact on H.323 behavior (rather than being applied on a per-interface basis). Global options include:

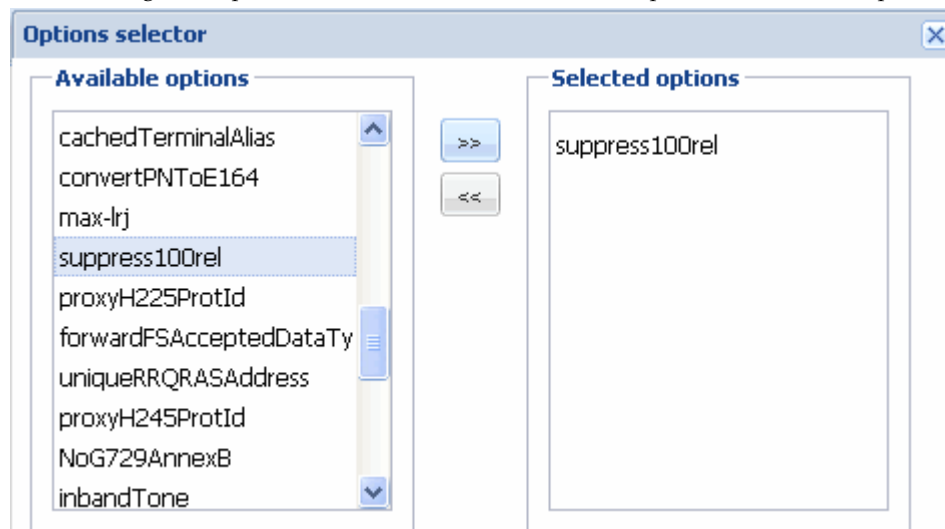
- **IwflRQ**: prevents the h323d from allocating separate ports for audio and T.38. After receiving a Request mode for T.38 and a fresh OLC for T.38, the Net-Net SBC sends the same audio port in the OLC ACK. This is done by sending the original m line to mbcd, instead of two m lines (one each for audio and T.38).
- **MapG729**: causes the Net-Net SBC to map H.245 g729 to SDP g729 w/annexb and vice versa. Applies only to H323-SIP IWF calls.
- **NoG729AnnexB**: the SDP received by the IWF with G.729 codec and no fmtp is mapped to G.729 on the H.323 side.
- **sameT38Port**: prevents the h323d from allocating separate ports for audio and T.38. After receiving a Request mode for T.38 and a fresh OLC for T.38, the Net-Net SBC sends the same audio port in the OLC ACK. This is done by sending the original m line to mbcd, instead of two m lines (one each for audio and T.38).

To configure global options:

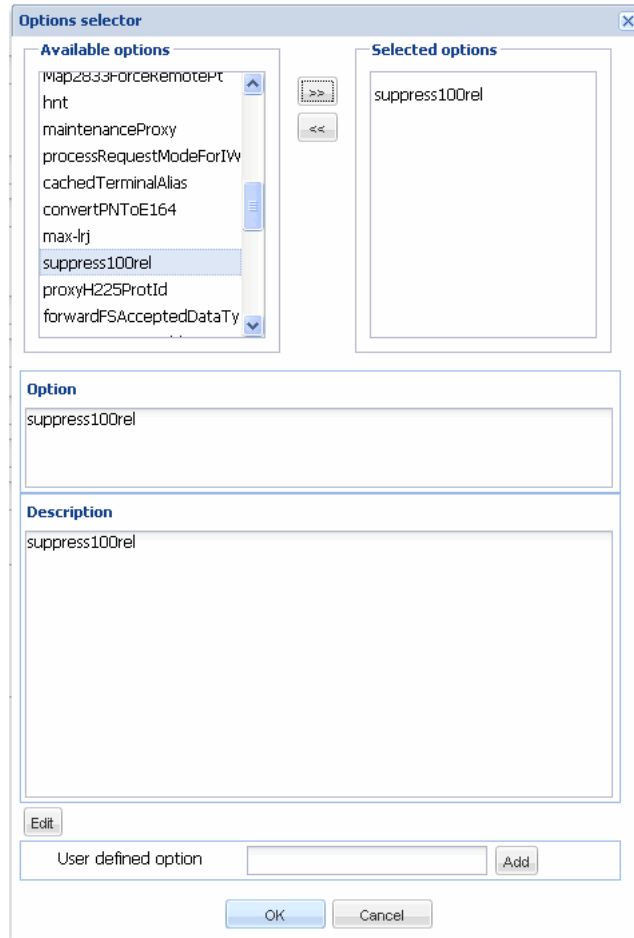
1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Scroll to Options and click **Options**.
5. Click a global option name to select it in the Select options list. For example.



6. Click  to move the option to the opposite list. The option name and details appear at the bottom of the screen.



The **Options selector** dialog box is shown. It contains two lists: **Available options** and **Selected options**. The **Available options** list includes: map2833-forceRemotePt, hnt, maintenanceProxy, processRequestModeForIw, cachedTerminalAlias, convertPNTtoE164, max-lrj, suppress100rel, proxyH225ProtId, and forwardFSAcceptedDataTy. The **Selected options** list contains: suppress100rel. Below the lists, the **Option** field displays 'suppress100rel' and the **Description** field also displays 'suppress100rel'. At the bottom, there is an **Edit** button, a text field labeled 'User defined option', an **Add** button, and **OK** and **Cancel** buttons.

7. Repeat steps 5 and 6 to add additional options.
8. Click **OK**.
9. Click **Apply**.

Configuring H.323 Interfaces

You need to configure H.323 interfaces for inbound and outbound traffic. When you configure H.323 interfaces, you can set:

- Identity and state
- Realm and H.323 interface associations
- H.323 interface settings for the interface's IP address, RAS and Q.931 ports, maximum number of Q.931 ports to allow, and any Annex E support you need
- H.323 system resource allocation

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by	admin@console
Last modified date	2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2.UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

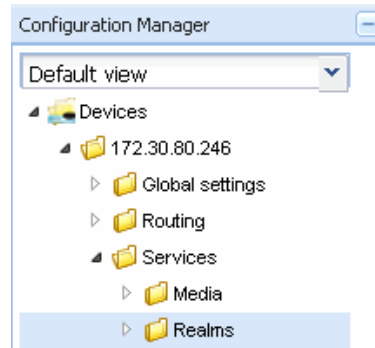
If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Configuring H.323 Interfaces

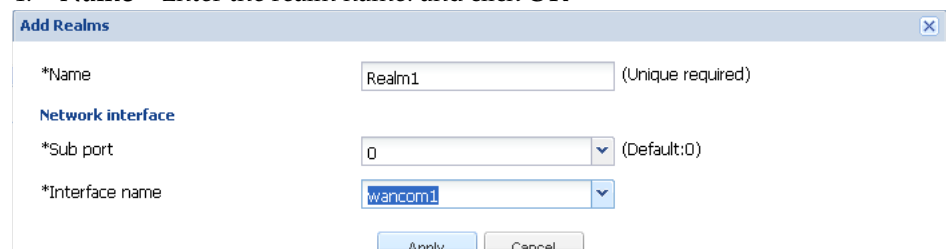
This section describes how to configure H.323 interfaces on your Net-Net SBC from the Net-Net Central.

Create the Realm

1. Expand the Configuration Manager slider > Services.

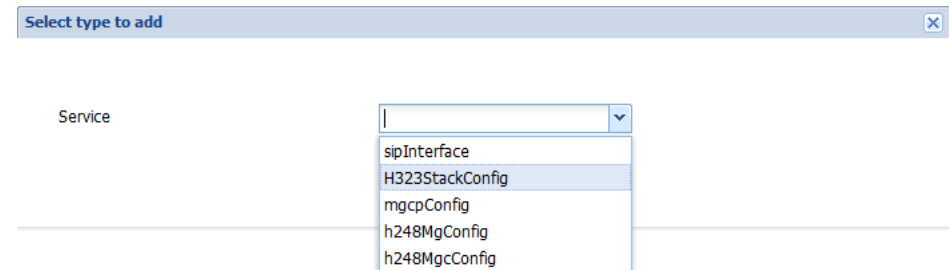


2. Click Realms. The Realm table appears in the content area.
3. Click **Add**. The Add Realm dialog box appears.
4. **Name**—Enter the realm name. and click **OK**



The new realm appears in the Realm table.

5. Select the new realm in the table and click **Edit**.
6. Click the Services/Protocols tab at the top of the content area.
7. Click **Add**. The Add Realm service window appears.
8. **Service/Protocol**—Select H323StackConfig from the combination box and click **Apply**.



The protocol is saved and the Name text box appears on the Add Realm service dialog box.

9. **Name**—Enter a unique identifier for this H.323 service applied to the realm.
10. **Realm**—Select the realm to which this service applies and click **Edit**.
11. Click **Apply**. The realm appears on the Realm Services table
12. Select the H.323 service from the Realm Services table and click **Edit**.

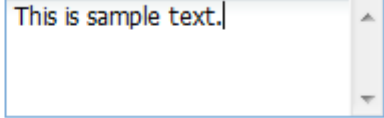
Configuring the Interface

You can set the following parameters to define basic settings for your H.323 interface. This is where you set the IP address for opening sockets, the RAS and Q.931 ports, and the maximum number of Q.931 calls that you want to allow.

This is also where you establish Annex E alternate transport. Annex E supports multiplexed call signaling over UDP so that call volume and performance are potentially enhanced. If you do not configure Annex E support, then this H.323 interface does not listen for Annex E requests.

1. **Description**—Enter an optional descriptive string to identify the configuration and its function.

Description



This is sample text.

2. **State**—Select enabled from the combination box.
3. **Outgoing H.323 service**—Select the outgoing H.323 interface from the combination box. Leaving this value *blank* is valid.

To use realm bridging with static routing, you need to set the outgoing H.323 interface. If you do not enter a name, the Net-Net SBC uses dynamic, policy-based selection using the local policy.

If you wish to use static stack selection, then each configured H.323 interface must have an associated outbound interface. This value must correspond to a valid name field value in another instance of the H.323 interface.

4. **IP address**—Enter the IP address that the H.323 interface uses when opening sockets; this is the default H.323 interface IP address. You must change this parameter from 0.0.0.0 to a valid IP address.
 - The IP address of a configured network interface must exist
 - The IP address must exist in the HIP list of that network interface
 - The physical interface the network interface is on must appear as a physical interface configured for the realm in which this H.323 stack exists
5. **RAS IP port-receive**—Enter the number of the port on the local IP address on which the Net-Net SBC listens for RAS requests. Acme Packet recommends you set this parameter to the well-known port 1719.

If you leave this parameter set to its default (0), the Net-Net SBC uses a port assigned by the operating system.

6. **Q931 port**—Enter the port number where to listen for Q.931 signaling messages. Acme Packet recommends you leave this parameter set to **1720**.

State	enabled
Outgoing H.323 service	InternalGK1

H.323 signaling interface

IP address	0.0.0.0
RAS IP port	1719
Q.931 port	1720

7. **Max. Q.931 concurrent calls (#)**—Enter the maximum number of concurrent Q.931 calls you want to allow. You can leave this parameter set to **200** (default). However, this value should be less than the maximum number of calls you set when configuring Fast Start/Slow Start Translations.

If the number of received Q.931 calls exceeds this number, the H.323 interface returns a *busy* state.

8. **Max. concurrent calls (#)**—Enter the maximum number of calls allowed on this H.323 stack.
9. **Max. channels (#)**—Enter the Maximum number of concurrent channels for each call on this H.323 interface.

Max. Q.931 concurrent calls	200

System resource allocation

Max. concurrent calls	200
Max. channels	6

The Net-Net SBC checks this parameter on initialization to reserve the appropriate network resources.

10. Click **Add** next to the Annex E alternate transport address: ports section. The Add Annex E transport address:port window appears.
11. **IP address Port**—Enter the IP address and port for alternate transport.
12. Click **Apply**.

Add Annex E alternate transport address:ports

*IP address:port	10.20.60.40:0	(Unique required. Enter IP and port in the format: x.x.x.x:0..65535. Default:0.0.0.0:0)
Apply Cancel		

If you do not configure this list, then the Net-Net SBC does not listen for incoming Annex E requests. You must enter the IP address and port combination.

13. Click **Apply**.

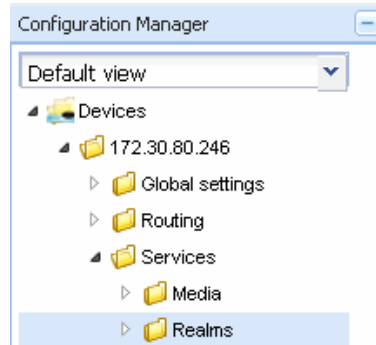
Configuring H.323 Service Modes

When you set the H.323 service mode, you configure parameters that define what type of service an H.323 interface provides. These parameters govern how the interface functions when you want it to behave as a gatekeeper or as a gateway.

Configuring the H.323 Service Mode

To configure the service mode for the H.323 interface:

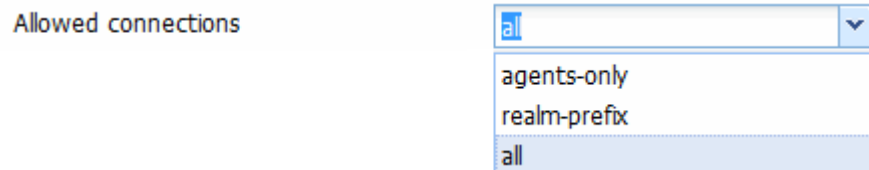
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Click the realm to select it from the Realm Services table and click **Edit**.
6. **Service mode**—Choose the service mode from the combination box.



7. **Allowed connections**—Choose which entities are considered trusted from the combination box.
 - **all**—Trust all previous and next hops except untrusted session agents
 - **agents-only**—Trust only trusted session agents
 - **realm-prefix**—Trust only trusted session agents or address matching realm prefix



8. **Gateway aliases identifiers**—Click **Add**.

H323-ID	Email	E164	Url

◀ ||| ▶

Add **Delete**

The H.323 gateway alias/gatekeeper ID dialog box appears. The value you enter here is either the gateway alias or the gatekeeper identifier, depending on the mode you configure for the interface. The aliases are set in the **sourceInfo** information element of outgoing ARQs. You must choose one field to use as an alias.

9. **Add Gateway aliases identifier**—Choose one alias type from the fields to use as an alias. Provided below are examples for each field.

Add Gateway aliases identifier [X]

H323-ID	xyz
Email	user@acmepacket.com
E164	17817566800
Url	http://acmepacket.com
IP address	63.67.143.4:2000

Apply **Cancel**

10. Click **Apply**. The new gateway alias identifier appears in the list.

Configuring Gateway Only Settings

If you are using the H.323 interface as a gateway, you might want to set registration time-out and address prefix parameters.

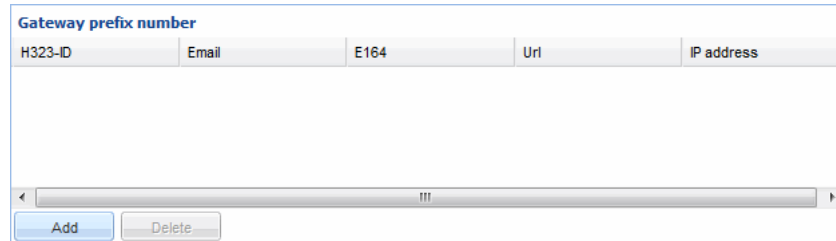
To configure gateway only settings:

1. **Registration expiration time-to-live (sec)**—Enter the number of seconds before a registration becomes invalid or leave the default value of **120**.

Registration expiration time-to-live (sec)

This value is used during the initial registration process. However, when a registration is confirmed, the time-to-live (TTL) value set by the gatekeeper in the Registration Confirm (RCF) message overrides this value.

2. **Gateway prefix numbers**—Click **Add** to enter a list of prefixes for this H.323 interface.

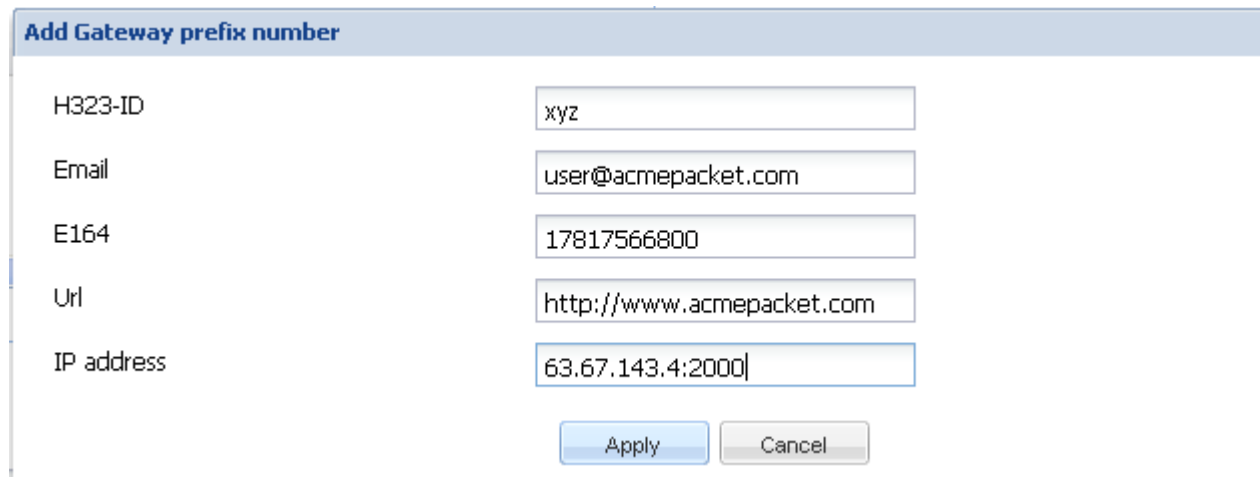


The dialog box titled "Gateway prefix number" contains a table with five columns: H323-ID, Email, E164, Url, and IP address. Below the table is a scrollable list area with an "Add" button and a "Delete" button.

The H.323 gateway prefix dialog box appears.

These prefixes are sent from a gateway interface to a gatekeeper and indicate valid prefixes accepted by that interface for incoming calls. They are used if the interface is configured as a gateway. You can enter one alias ID from the dialog box.

3. **Add Gateway prefix number**—Choose one alias type from the fields to use as an alias. Provided below are examples for each field.



The "Add Gateway prefix number" dialog box shows input fields for each column: H323-ID (xyz), Email (user@acmepacket.com), E164 (17817566800), Url (http://www.acmepacket.com), and IP address (63.67.143.4:2000). At the bottom are "Apply" and "Cancel" buttons.

4. Click **Apply**. The new gateway alias prefix appears in the list.

Gatekeeper Proxy Settings

If you are using the H.323 interface as a gatekeeper proxy, you might want to set:

- Whether registration processing is enabled or disabled
- Whether or not this H.323 interface is signaling-only
- At what H.225 call stage the H.245 procedures should be initiated

To configure gatekeeper proxy settings:

1. **Service mode**—Choose gatekeeper as the service mode.

Service mode	enabled
Allowed connections	enabled disabled

2. Scroll to Gatekeeper proxy only.
3. **Registration processing**—Retain the default value of disabled if you want the Net-Net SBC to drop all RRQs, meaning that it not acknowledges any requests. Choose enabled from the combination box if you want the Net-Net SBC will process any RRQs that arrive on this H.323 interface.

Registration processing	disabled
	enabled disabled

When registration processing is enabled and the Net-Net SBC receives an RRQ on this H.323 interface, it will route the request to the appropriate gatekeeper. After the gatekeeper confirms that registration with an RCF, the Net-Net SBC also confirms it with the endpoint that sent the RRQ. Then the registration becomes part of the Net-Net SBC's registration cache. If this endpoint does not confirm the registration, then the Net-Net SBC will reject the registration with an RRJ and will not cache it.

4. **Signaling proxy mode (no media)**—Select the proxy mode that you want to use for the signaling only operation mode from the combination box.

Signaling proxy mode (no media)	
H.225 call stage to initiate H.245 procedures	H245 H225

For more information about the H.225 or H.245 proxy mode, refer to the *Configuring H.323 Signaling Only Operation* description.

5. **H.225 call stage to initiate H.245 procedures**—Select the stage at which the Net-Net SBC transfers the H.245 address to the remote side of the call, or acts on the H.245 address sent by the remote side.

H.225 call stage to initiate H.245 procedures	connect
Other Options	dynamic proceeding connect facility setup alerting early noh245
H.245 tunneling	
RFC 2833	
DTMF RFC2833 mode	

For more information, refer to the *Dynamic H.245 Stage Support* description at the beginning of this chapter.

6. Click **Apply**.

Fast Start/Slow Start Translations

The Net-Net SBC can translate between Fast Start H.323 endpoints and Slow Start H.323 endpoints. Using this feature, you can reduce delay in establishing media, improve performance, and reduce network congestion caused by a high number of messages being exchanged. Fast Start and Slow Start calls handle information about media for a session in different ways. In a Fast Start call, information about the media is contained in the Setup message. In a Slow Start call, that information is exchanged between endpoints after the session has been established.

When you Fast Start/Slow Start translation, the Net-Net SBC can take a Slow Start call from an H.323 endpoint that does not support Fast Start and re-initiate that call as Fast Start. It also allows an H.323 endpoint that does not support Fast Start to receive a Slow Start call from a Fast Start source because the Net-Net SBC performs all necessary translations.

Note: When you configure an H.323 interface, you configure it for either Fast Start to Slow Start translation or for Slow Start to Fast Start translation. You cannot configure one H.323 interface for both translation modes.

Fast Start to Slow Start Translation

The Net-Net SBC supports translations from H.323 Fast Start to Slow Start. Using this feature, an H.323 endpoint that only supports Slow Start can call from a Fast Start source when that call goes through the Net-Net SBC.

In a Fast Start call, the originating H.323 endpoint sends a **fastStart** element in its Setup message. This element contains H.245 OLC messages that allow Fast Start endpoints to establish a media stream when the call is connected. As a result fewer messages are exchanged between the H.323 endpoints than there would be for a Slow Start call (where the **fastStart** element does not appear). Because media information is sent in the Setup request for the session, there is no need to use the media profiles when converting a Fast Start call to Slow Start.

When you enable the slow start option in the H.323 stack configuration, the Net-Net SBC performs Fast Start to Slow Start conversion. During the translation, the Net-Net SBC retains the media information included in the incoming Fast Start call as it negotiates a connection with the Slow Start endpoint. After a connection with the Slow Start endpoint has been established, the Net-Net SBC negotiates the media capabilities.

Slow Start to Fast Start Translation

When you configure your Net-Net SBC to support H.323 Slow Start to Fast Start translations, you enable an H.323 endpoint that only supports Slow Start to initiate and sustain communication with an H.323 Fast Start endpoint. The Net-Net SBC resolves the Slow Start limitation of exchanging information about media (OLC messages) after the call is connected. The OLC message opens a logical channel, or a unidirectional or bi-directional path used to transmit media packets. Using the Net-Net SBC, you can negotiate the construction of media flows differently, which is described in this section.

When you enable the Fast Start option for calls in the H.323 stack configuration, the Net-Net SBC performs the translation of a Slow Start call into Fast Start. When it receives a Slow Start call, the Net-Net SBC determines its destination and the H.323 stack it uses for the outgoing call.

It is a requirement of this kind of translation that you configure and use media profiles. Since a Slow Start call does not negotiate media until after the call is connected, there needs to be an assumption made about the media to set up a Slow Start to Fast Start call. Media profiles fill this role, and they are assumed to be part of a correct configuration.

The following describes possible scenarios for Slow Start to Fast Start translations.

- When a Slow Start call arrives at the Net-Net SBC and matches one of the session agents that has a media profiles list configured, the outgoing call is set up as a Fast Start call. The session agent's media profiles are used for the logical channels. You must configure the media profiles to reference a codec the endpoint accepts.

If there are no media profiles configured for the session agent, then the Net-Net SBC uses the media profiles list in the H.323 stack configuration to open the logical channels.

- If a Slow Start call arrives at the Net-Net SBC and its destination does not match one of the session agents, the Net-Net SBC uses the media profiles list in the H.323 stack configuration for the outgoing call. If there is a list of media profiles, the outgoing call is set up as a Fast Start call with the media profiles list used to open the logical channels.

If there is no list of media profiles for the outgoing H.323 interface, the Net-Net SBC does not perform Slow Start to Fast Start translation. The Slow Start call exits the Net-Net SBC as it arrived—as a Slow Start call.

- If the egress H.323 interface has the Fast Start option disabled, then the outgoing call uses the Slow Start mode, and the Net-Net SBC does not perform Slow Start to Fast Start translation. In this case, the Slow Start call also exits the Net-Net SBC as it arrived—as a Slow Start call.

Configuration Prerequisites

To perform Fast Start/Slow Start translations, you need to have a standard two-interface configuration already in place.

If you are using the Slow Start to Fast Start translations, you must configure appropriate entries in the media profiles list which is part of the translation parameters. Ensure that you use the name of a configured media profile when you enter values in the media profiles list.

Note: When you configure an H.323 interface, you configure it for either Fast Start to Slow Start translation or for Slow Start to Fast Start translation. You cannot configure one H.323 interface for both translation modes.

Creating Media Profiles

Configure a list of media profiles if you want the local policy to route SIP and H.323 traffic by the codecs specified in the SDP. The list of media profiles entered here are matched against the SDP included in SIP or H.323 requests and the next hop is selected by codec.

The values in this list are matched against the rtpmap attribute of passed SDP, and preference weight for route selection is based on the order in which the matching payload type appears in the SDP's media (m=) line.

For example when the following SDP arrives:

```
m=audio 1234 RTP/AVP 0 8 18
```

that contains the following attributes that correspond to three configured local policies with the same cost:

- `a=rtpmap:0 PCMU/8000`
- `a=rtpmap:8 PCMA/8000`
- `a=rtpmap:18 G729/8000`

the following route selection action occurs:

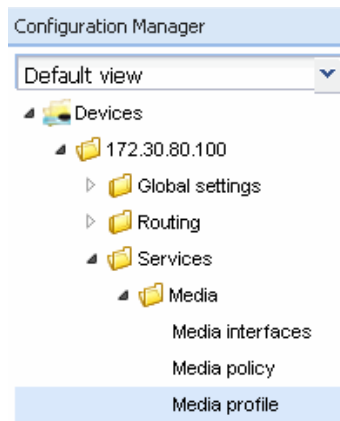
The local policy route that corresponds to the `a=rtpmap:0 PCMU/8000` attribute is selected because the payload type of 0 in the attribute line matches the first payload type of 0 listed in the `m=` line. The codec value of PCMU indicated in this selected attribute is used to find the local policy with the media profiles attribute that includes PCMU in the list.

Because the value you enter here is matched against the codec values included in the actual passed SDP, it must correspond to accepted industry-standard codec values.

Configuring Media Profiles

To create a media profile:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**.

4. **Media profile name**—Select from the combination box.

5. **Subname**—Enter a subname to create multiple media profiles with the same codec name; using a bandwidth value is convenient. For example, you might set a subname of 64k for media profile with **name** value of PCMU
6. Click **OK**. The new media profile appears in the media profile table.
7. Click the new media profile in the media profile table and click **Edit**. The Edit Media profile window appears.
8. **Media type**—Choose the type from the combination box.

Media type

9. **Payload type**—Enter the payload type number. This field is used by the system to determine the encoding type when the SDP included with a session identifies the standard payload type on the m line, but does not include an a-rtpmap entry.
- For RTP/AVP media-profile elements, this field should only be configured when there is a standard payload type number that corresponds to the encoding name. No payload type number is assigned for newer, dynamic codecs.
10. **Transport protocol**—Select the transport protocol from the combination box.
11. **Frames per RTP packet**—Enter the maximum number of frames to use per RTP packet, which facilitates Slow Start translations to Fast Start. Min: 0 / Max: 256. A value of 0 means that this field is not being used.
12. **Bandwidth requirements (Kbps)**—Enter the total bandwidth that the media requires in kilobits per second. Min: 0 / Max: $2^{32}-1$.

13. **Peak rate limit (bytes/sec)**—Enter the peak rate limit in bytes per second. This rate limit is the flowspec parameter r (bucket rate) / p (peak rate) value to insert into COPS message for RACF/PDP configuration.
14. **Maximum burst size (bytes)**—Enter the maximum burst size in bytes. This burst size is the flowspec parameter b (bucket depth) / m (minimum policed unit) / M (maximum datagram size) value to insert into COPS message for RACF/PDP configuration.
15. **Media policing**—Select enabled from the combination box too allow media flow policing.

Media type	audio ▼
Payload type	0
Transport protocol	RTP/AVP ▼
Frames per RTP packet	1
Bandwidth requirements (Kbps)	120
Parameters	
Peak rate limit (bytes per second)	0
Maximum burst size (bytes)	0

16. **Maximum bandwidth allowed per flow (bytes/sec)**—Enter the sustained rate limit in bytes per second for the media flow, with a maximum of 125Mbytes per second.

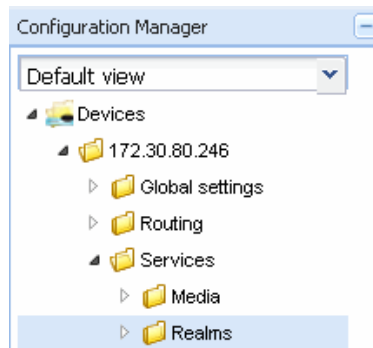
Maximum bandwidth allowed per flow (bytes per second),Media policing	3600

17. Click **Apply**.

Configuring Translations

When you configure an H.323 interface, you configure it for either Fast Start to Slow Start translation or for Slow Start to Fast Start translation. You cannot configure one H.323 interface for both translation modes.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 stack service in the table and click **Edit**.
6. Click the H.323 interworking tab at the top of the content area.
7. **Translate incoming slow start to outgoing fast start call**—Select enabled from the combination box if you want the Net-Net SBC to translate Slow Start calls to Fast Start calls.

or

Translate incoming fast start to outgoing slow start call—Select enabled from the combination box if you want the Net-Net SBC to translate Fast Start calls to Slow Start calls.

8. **Respond to fast start SETUP using fast start fields**—Select enabled from the combination box if you want the Net-Net SBC to include Fast Start fields in the response to the first message.

Translate incoming slow start to outgoing fast start call enabled 

Media profiles for outgoing fast start call

Name

Add Delete

Translate incoming fast start to outgoing slow start call

disabled 

Respond to fast start SETUP using fast start fields

disabled 

9. **Media profiles for outgoing fast start call**—Click the **Add** button in the Media Profiles section to add a media profile for an outgoing fast start call.

Media profiles for outgoing fast start call

Name

Add Delete

The Add media profiles for outgoing fast start call window appears.

10. **Media profile**—Select the media profile you want to use when translating Slow Start calls to Fast Start from the combination box. This information is used to open logical channels for the outgoing call.

Add Media profiles for outgoing fast start call 

*Media profile telephone-event  (Unique required)

- telephone-event
- G722
- G723
- G726-16
- G729A
- GSM

If you enter the name of a media profile that does not exist, the Net-Net SBC will not perform translation. If you leave this parameter empty, the Net-Net SBC will not perform translation.

11. Click **Apply** to continue. The media profile name appears in the Media profiles for outgoing fast start call list.
12. **Respond to fast start SETUP using fast start fields**—Select enabled from the combination box if you want to include Fast Start fields in the first message that the Net-Net SBC uses to respond to a Setup message.
13. Click **Apply**.

RFC 2833: DTMF Interworking

This section explains the Net-Net SBC's support of transporting Dual Tone Multi-Frequency (DTMF) in Real-Time Transport Protocol (RTP) packets (as described in RFC 2833) to H.245 User Input Indication (UII) or SIP INFO method interworking.

Multimedia devices and applications must exchange user-input DTMF information end-to-end over IP networks. The Net-Net SBC provides the interworking capabilities required to interconnect networks that use different signaling protocols. Also, the Net-Net SBC provides DTMF translation to communicate DTMF across network boundaries.

The Net-Net SBC supports RFC 2833 to H.245 UII translation for H.323-to-H.323 calls, when one side is a version 4 H.323 device requiring RFC-2833 DTMF event packets, and the other side is a pre-version 4 H.323 device that only uses H.245 UII.

About RFC 2833

RFC 2833 specifies a way of encoding DTMF signaling in RTP streams. It does not encode the audio of the tone itself, instead a signal indicates the tone is being sent. RFC 2833 defines how to carry DTMF events in RTP packets. It defines a payload format for carrying DTMF digits used when a gateway detects DTMF on the incoming messages and sends the RTP payload instead of regular audio packets.

About H.245 UII

H.245 provides a capability exchange functionality to allow the negotiation of capabilities and to identify a set of features common to both endpoints. The media and data flows are organized in logical channels. H.245 provides logical channel signaling to allow logical channel open/close and parameter exchange operations. The H.245 signaling protocol is reliable, which ensures that the DTMF tones will be delivered.

H.245 User Input Indication (UII) plays a key role in all the services that require user interaction. For video messaging, typical uses of UII include selection of user preferences, message recording and retrieval, and typical mailbox management functions. H.245 UII provides two levels of UII, alphanumeric and signal.

About 2833 to H.245 UII Interworking

The Net-Net SBC provides 2833 to H.245-UII interworking by checking 2833-enabled RTP streams for packets matching the payload type number for 2833. It then sends the captured packet to the host for processing and translation to H.245 UII messages. A H.245 UII message received by the Net-Net SBC is translated to 2833 packets and inserted into the appropriate RTP stream.

About DTMF Transfer

DTMF transfer is the communication of DTMF across network boundaries. It is widely used in applications such as interactive voice response (IVR) and calling card applications.

The multiple ways to convey DTMF information for packet-based communications include:

- In-band audio: DTMF digit waveforms are encoded the same as voice packets. This method is unreliable for compressed codecs such as G.729 and G.723
- Out-of-band signaling events:
 - H.245 defines out-of-band signaling events (UII) for transmitting DTMF information. The H.245 signal or H.245 alphanumeric methods separate DTMF digits from the voice stream and send them through the H.245 signaling channel instead of through the RTP channel. The tones are transported in H.245 UII messages.
 - All H.323 version 2 compliant systems are required to support the H.245 alphanumeric method, while support of the H.245 signal method is optional.
 - SIP INFO uses the SIP INFO method to generate DTMF tones on the telephony call leg. The SIP INFO message is sent along the signaling path of the call. Upon receipt of a SIP INFO message with DTMF content, the gateway generates the specified DTMF tone on the telephony end of the call.
- RTP named telephony events (NTE): uses NTE to relay DTMF tones, which provides a standardized means of transporting DTMF tones in RTP packets according to section 3 of RFC 2833.

Of the three RTP payload formats available, the Net-Net SBC supports RTP NTE. NTE is most widely used for SIP devices but is also supported in H.323 version 4 or higher endpoints.

RFC 2833 defines the format of NTE RTP packets used to transport DTMF digits, hookflash, and other telephony events between two peer endpoints. With the NTE method, the endpoints perform per-call negotiation of the DTMF transfer method. They also negotiate to determine the payload type value for the NTE RTP packets.

The NTE payload takes the place of codec data in a standard RTP packet. The payload type number field of the RTP packet header identifies the contents as 2833 NTE. The payload type number is negotiated per call. The local device sends the payload type number to use for 2833 telephone event packets using a SDP or H.245 Terminal Capability Set (TCS), which tells the other side what payload type number to use when sending the named event packets to the local device. Most devices use payload type number 101 for 2833 packets, although no default is specified in the standard.

The 2833 packet's RTP header also makes use of the timestamp field. Because events often last longer than the 2833 packets sending interval, the timestamp of the first 2833 packet an event represents the beginning reference time for subsequent 2833 packets for that same event. For events that span multiple RTP packets, the RTP timestamp identifies the beginning of the event. As a result, several RTP packets might carry the same timestamp.

See RFC 2833 and draft-ietf-avt-rfc2833bis-07.txt for more information.

Preferred and Transparent 2833

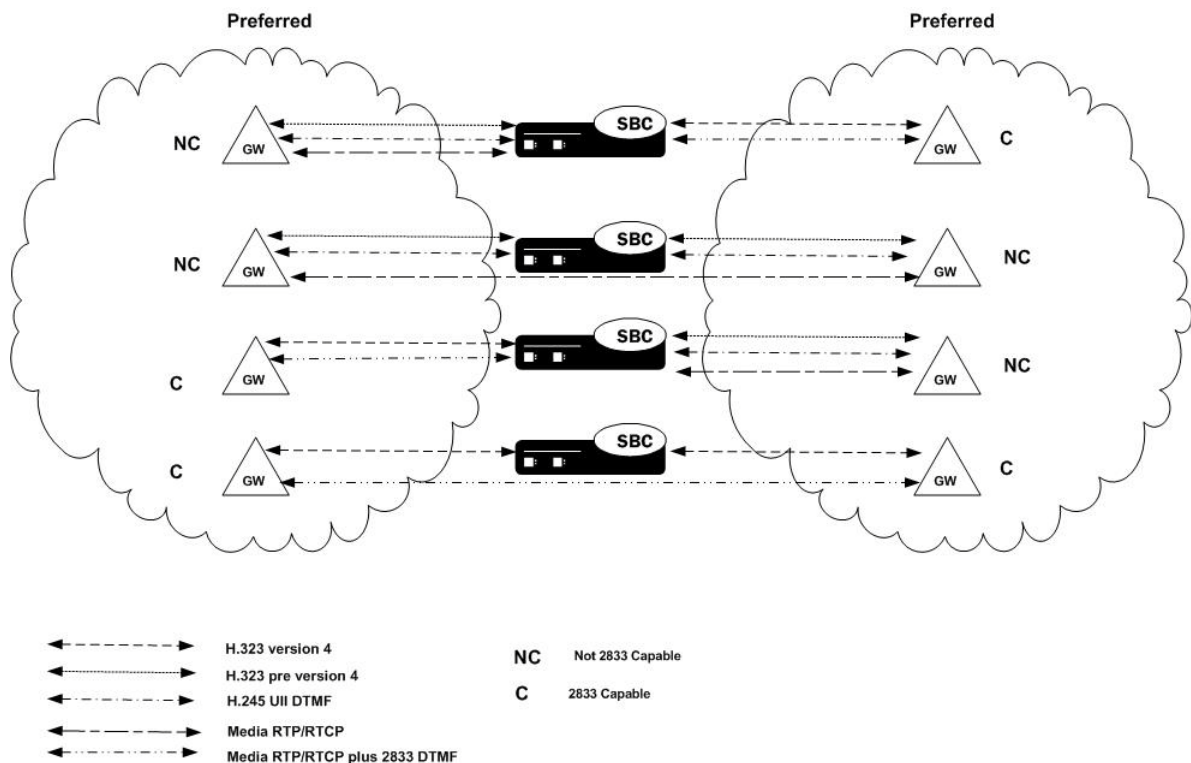
To support preferred (signaled) 2833 and transparent 2833, the Net-Net SBC provides 2833 detection and generation (if necessary) when the endpoint signals support for 2833.

- Preferred: the Net-Net SBC only generates and detects 2833 for endpoints if they negotiate support for 2833 through signaling
- Transparent: the Net-Net SBC behaves as it has prior to this release, offering and answering based on end-to-end signaling and transparently relaying 2833

Preferred 2883 Support

If one side of the call, or a session agent, is configured for preferred 2833, the Net-Net SBC only generates and detects 2833 for endpoints if they signal support for 2833. The Net-Net SBC will offer 2833 in the TCS SDP, even if the originating caller did not.

- When the Net-Net SBC manages calls originating from a preferred source going to a preferred target, it:
 - Performs 2833 translation for an endpoint when the originating side requests 2833 but the target does not negotiate 2833
 - Allows 2833 to pass through if the originating side and target of the call are configured as preferred and negotiate 2833
- When the Net-Net SBC manages calls originating from a preferred source going to a transparent target, it:
 - Performs 2833 translation when the originating side requests 2833 but the target is configured as transparent and does not negotiate 2833.
 - Allows 2833 to pass through if the originating side and the target of the call are configured as transparent and negotiate 2833. The Net-Net SBC does not perform active translation because both ends support 2833.



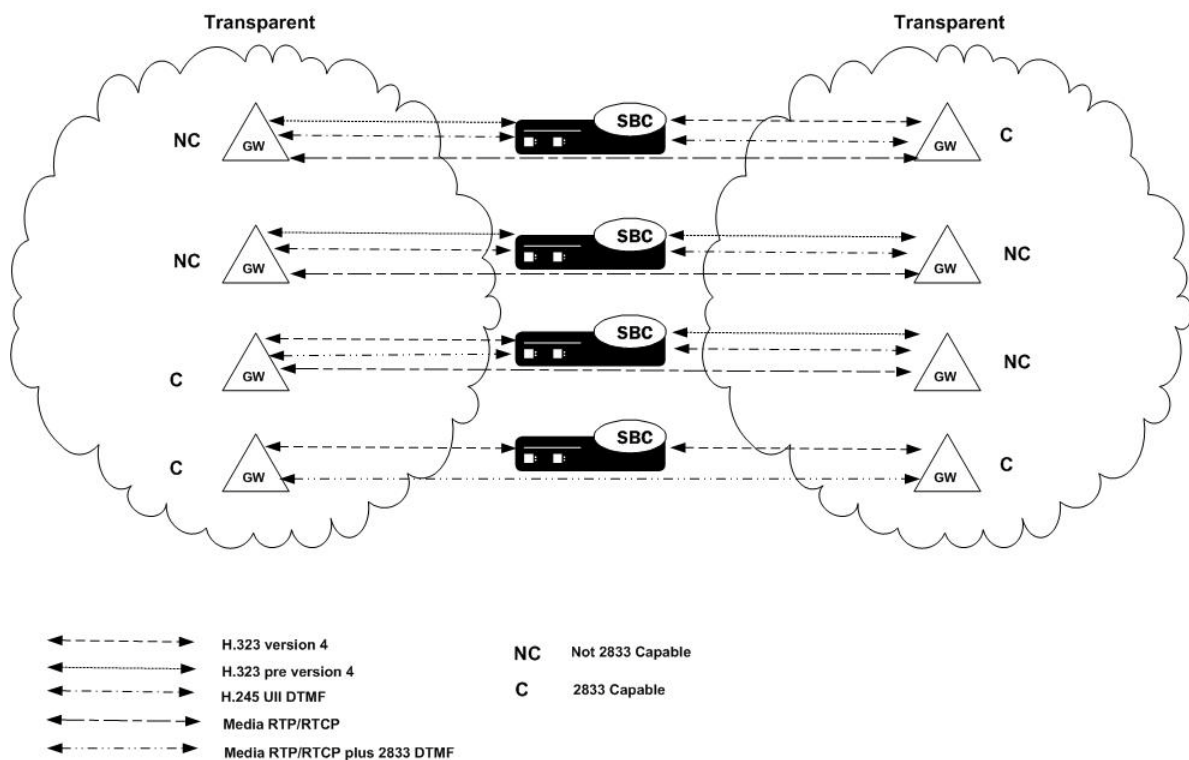
If one SIP endpoint does not signal 2833 capability, but the other SIP or H.323 endpoints do, the Net-Net SBC does not perform 2833 translation.

Transparent 2833 Support

The default configuration of the Net-Net SBC for H.323 is transparent 2833. The Net-Net SBC passes on the offered capabilities to the next-hop signaling element. If the next-hop endpoint is for a transparent 2833 target, typical capability negotiation determines the DTMF method. The Net-Net SBC transparently relays the DTMF as it has in previous releases.

With transparent 2833, the Net-Net SBC acts as a typical B2BUA or B2BGW/GK. However when the target of the call is configured as preferred 2833, the Net-Net SBC:

- Relays the 2833 packets if the originating endpoint signals 2833 and the next-hop endpoint for the preferred target signals 2833
- Performs 2833 translation if the originating endpoint does not signal 2833 and the next-hop endpoint for the preferred target does signal 2833
- Does not perform 2833 translation or transparently relay 2833 if the originating endpoint signals 2833 and the next-hop endpoint for the preferred target (or even a transparent 2833 target) does not signal 2833.



Basic RFC 2833 Negotiation Support

If H.323, SIP, or session agents on either side of the call are configured for preferred 2833 support, the Net-Net SBC supports end-to-end signaled negotiation of DTMF on a call-by-call basis. If the calling party is not configured for preferred support but sends 2833, the Net-Net SBC sends 2833 to the next-hop called party. If the calling party sends H.245 signals or alphanumeric UII, the Net-Net SBC sends H.245 signals or alphanumeric UII to the next-hop called party (if it is an H.323 next-hop).

The Net-Net SBC also supports hop-by-hop negotiation of DTMF capability on a call-by-call basis, if the signaling protocols or session agents on either side of the call are configured for preferred 2833 support.

H.323 to H.323 Negotiation

The Net-Net SBC serves as the H.323 called gateway. It answers RFC 2833 audio telephony event capability in the version 4 H.323/H.245 TCS when it receives a call from an H.323 endpoint configured for preferred RFC 2833.

If the Net-Net SBC is the answering device, configured for preferred support, and the calling device sends 2833, the Net-Net SBC accepts the 2833 regardless of the next-hop's DTMF capabilities. The received dynamic RTP payload type is used for detecting 2833 packets, while the response dynamic payload type is used for generating 2833 packets.

The Net-Net SBC supports:

- RFC-2833 audio telephony events in the version 4 H.323/H.245 TCS as the H.323 calling gateway, when the Net-Net SBC calls an H.323 endpoint configured for preferred RFC 2833 support. The Net-Net SBC sends 2833 to the called party regardless of whether the calling party sends it.
- H.245 UII and RFC-2833 packets sent at the same time, to the same endpoint, even if only half of the call is being provided 2833 support by the Net-Net SBC.

If one half of the call supports H.245 UII, and the other half is being provided 2833 translation by the Net-Net SBC, the Net-Net SBC can also forward the H.245 UII it receives to the 2833 endpoint. For example, when the signaling goes through a gatekeeper or third party call control, sending the H.245 UII in the signaling path allows those devices to learn the DTMF digits pressed.

Signal and Alpha Type Support

The Net-Net SBC supports:

- H.245 signal and alpha type UII in the H.323/H.245 TCS as the H.323 calling gateway when the:
 - Net-Net SBC calls an H.323 endpoint configured for transparent 2833 support
 - calling endpoint's target is configured as preferred

If the originating preferred side also sends 2833, the Net-Net SBC forwards it to the transparent side. The Net-Net SBC sends signal and alpha UII support to the called party regardless of whether the calling party sends it, if the call originates from a preferred side to a transparent side.

- H.245 alphanumeric UII for DTMF for H.323 endpoints that do not signal 2833 or contain explicit H.245 UII capability, for stacks configured for transparent 2833 support.

When the other half of the call is an H.323 endpoint of a stack configured for preferred 2833, the Net-Net SBC translates incoming H.245 UII on the transparent side, to 2833 packets on the preferred side, and vice versa. If the other half of the call is an H.323 endpoint of a transparent stack, the Net-Net SBC relays the H.245 UII messages.

- H.245 signal type UII for DTMF for H.323 endpoints that do not signal 2833, but do signal explicit H.245 UII capability, for stacks configured for transparent 2833 support.

When the other half of the call is an H.323 endpoint of a stack configured for preferred 2833, the Net-Net SBC translates incoming H.245 signaled UII on the transparent side, to 2833 packets on the preferred side, and vice versa. If the other half of the call is an H.323 endpoint of a transparent stack, the Net-Net SBC relays the H.245 UII messages if both sides support it.

H.323 Endpoints

Because there are different H.323 endpoints based on different versions of H.323, the DTMF can be either be transferred out-of-band as UII or in-band using RFC 2833. Most H.323 endpoints:

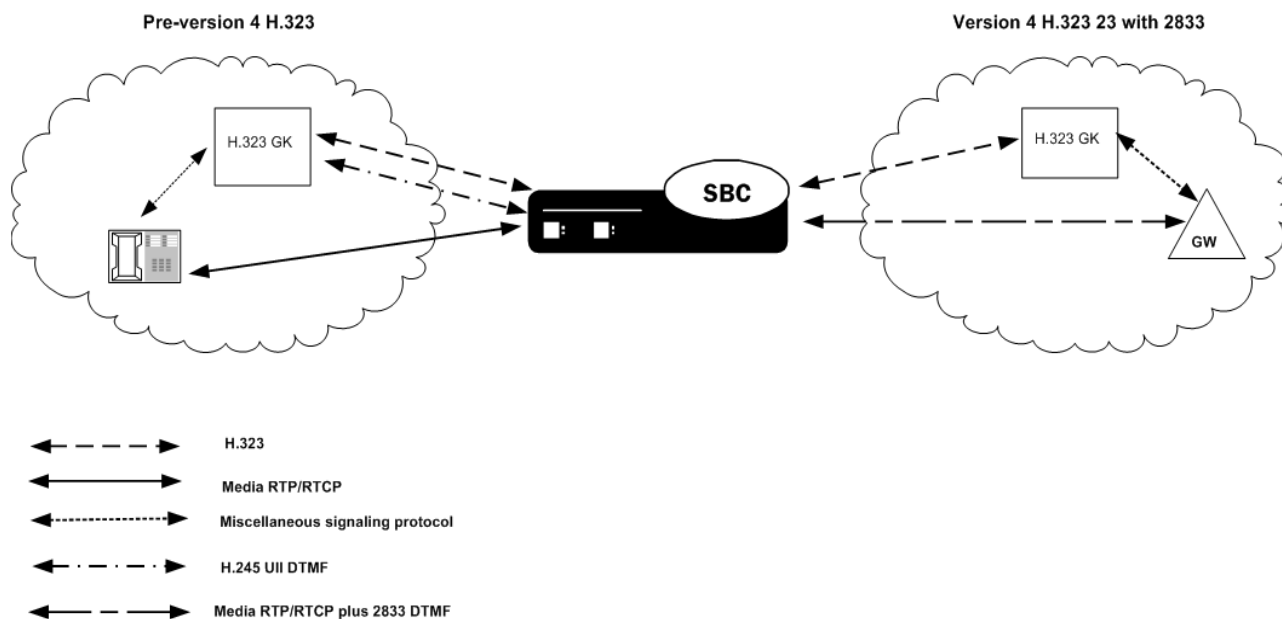
- version 4 and above support RFC 2833
- version 2 and pre-version 4 support UII-Signal
- version 1 and pre-version 2 support UII-Alphanumeric

Translating H.245 UII to 2833 for H.323 Calls

A majority of H.323 endpoints are not version 4 H.323 compliant and do not support RFC 2833 for DTMF transfer. However, some networks include version 4 H.323 devices that require the DTMF events to be signaled in 2833 packets. Network-based version 4 H.323 gateways use RFC 2833 instead of H.245 UII. (Version 4 H.323 devices should support H.245 UII.)

The Net-Net SBC translates 2833 to H.245 UII for H.323-to-H.323 calls when one side is a version 4 H.323 device requiring RFC-2833 DTMF event packets, and the other side is a pre-version 4 H.323 device which only uses H.245 UII.

The Net-Net SBC can translate H.245 UII to RFC2833 and back, based on the admin configuration and H.245 TCS exchanges. This translation enables DTMF to work end-to-end.



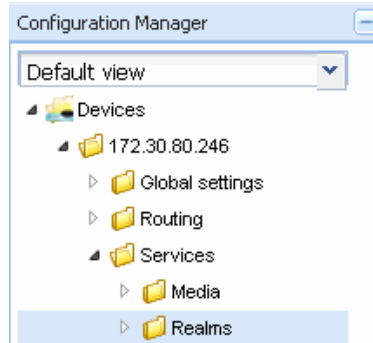
Configuring DTMF Support

This section describes how to configure DTMF support on your Net-Net SBC from the Net-Net Central.

RFC 2833 Mode for H.323 Interfaces

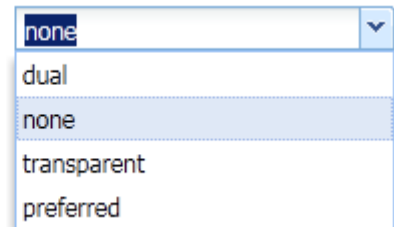
To configure RFC 2833 mode for H.323 interfaces:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab at the top of the content area.
7. **DTMF RFC2833 mode**—Choose the RFC 2833 mode from the combination box.

DTMF RFC2833 mode

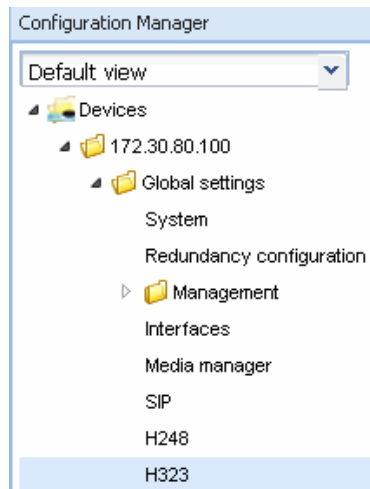


- **transparent**—2833 or UII negotiation is transparent to the Net-Net SBC.
 - **preferred**—The H323 stack uses 2833 for DTMF transfer, which it signals in its TCS. However, the remote H323 endpoint makes the decision. If the endpoint supports 2833, 2833 is used. If not, the H.323 stack reverts back to using UII.
8. Click **Apply**.

RFC 2833 Payload for H.323

To configure the RFC 2833 payload preferred mode:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. **Payload for RFC2833 packets interworked to UII**—Enter a number that indicates the payload type the Net-Net SBC will use for RFC 2833 packets while interworking 2833 and UII. Value range is 96 to 127, with the default value of 101.

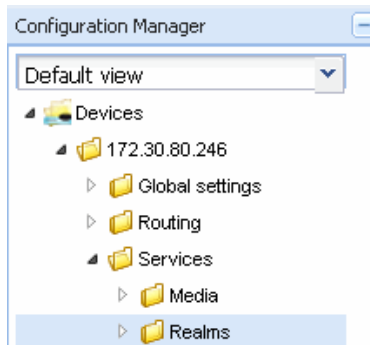
Payload type for RFC2833 packets
interworked to UII

5. Click **Apply**.

Configuring the SIP Interface

You configure the 2833 mode and payload for the SIP interface. You must configure the payload the Net-Net SBC will use for RFC 2833 packets, while interworking 2833 and INFO/UII.

1. Expand the Configuration Manager slider > Services.



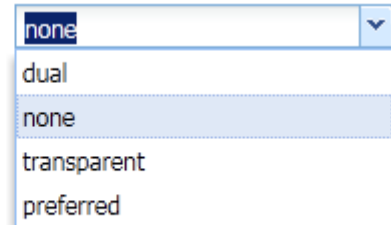
2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.

5. Select a SIP interface service in the table and click **Edit**.
6. **Payload type for RFC2833 packets interworked to UII**—Enter a number that indicates the payload type the SIP interface will use for RFC 2833 packets while interworking 2833 and UII. Value range is 0 and 96 to 127.

Payload type for RFC2833 packets
interworked to UII

7. **DTMF RFC2833 mode**—Choose the RFC 2833 mode from the combination box.

DTMF RFC2833 mode



none
dual
none
transparent
preferred

- **transparent**—2833 or UII negotiation is transparent to the Net-Net SBC.
 - **preferred**—SIP interface uses 2833 for DTMF transfer.
8. Click **Apply** to continue.

Configuring Session Agents

You configure session agents with:

- Payload type the Net-Net SBC wants to use for RFC 2833 packets while interworking 2833 and UII.

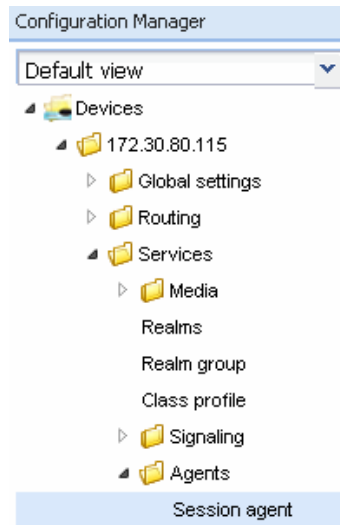
The default value for this attribute is 0. When this value is zero, the global rfc2833-payload configured in the h323-configuration element will be used instead. For SIP session agents, the payload defined in the SIP interface is used, if the SIP interface is configured with the preferred RFC 2833 mode.

- 2833 mode

A value of transparent or preferred for the session agent's 2833 mode will override any configuration in the h323-stack configuration element.

To configure session agents:

1. Expand the Configuration Manager slider > Services > Agents.

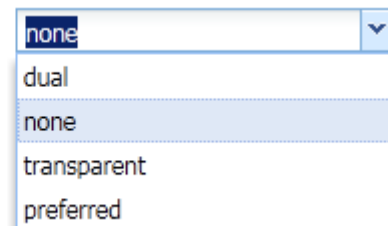


2. Click Session agent. The Session agent table appears in the content area.
3. Select the session agent for which you want to configure RFC 2833 to H.245 User UII or SIP INFO method interworking to select it and click **Edit**.
4. Click Translation and manipulation tab at the top of the content area.
5. **Payload type for RFC2833 packets interworked to UII**—Enter a number that indicates the payload type the session agent will use for RFC 2833 packets while interworking 2833 and UII. Value range is 0 and 96 to 127.

Payload type for RFC2833 packets
interworked to UII

6. **DTMF RFC2833 mode**—Select the RFC 2833 mode from the combination box.

DTMF RFC2833 mode



7. Click **Apply**.

H.323 Registration Proxy

The Net-Net SBC provides a registration proxy feature that allows a gatekeeper to authenticate a registration before accepting it. This feature is key when two factors are present: authentication is required, and an RRQ from an endpoint includes a token and/or cryptographic token. If authentication for that endpoint is to work, the Net-Net SBC must forward the registration requests received from the endpoint to the gatekeeper separately. When you do not use the H.323 registration proxy, the Net-Net SBC combines all registrations received from H.323 endpoints into a single RRQ and sends it to the gatekeeper. Using the H.323 registration proxy, you can configure the Net-Net SBC to use separate forwarding.

When registration requests are forwarded separately, each RRQ must have a unique CSA. This means that the Net-Net SBC must perform a one-to-one translation of the CSA in the incoming RRQ to a distinct transport address. The translated address replaces the endpoint's CSA in the outgoing RRQ. Then the Net-Net must listen for incoming calls that arrive at this translated transport address for the registered endpoint.

H.235 Authentication Transparency

When operating in this mode, H.235 authentication tokens (cryptotokens) in RAS messages proxied through the Net-Net SBC are passed through transparently.

For applications where Net-Net SBC is between H.323 gateways and a network hosted gatekeeper, the H.235 cryptotokens are passed through unmodified in RAS messages: RRQs, ARQs, and DRQs. This feature allows for secure gateway authentication.

Unique CSA Per Registered Gateway

When operating in this mode, each CSA is mapped to a registered gateway for call routing. The core gatekeeper does not support additive registrations, so a different CSA must be used for each unique registration that goes to the gatekeeper. The gatekeeper does not overwrite previously registered aliases. Also, since the gatekeeper initiates calls to an endpoint on the CSA specified in the RRQ, the Net-Net SBC must listen on the assigned address for incoming calls to that client as long as the client is registered.

Virtual Call Signaling Address

You can configure the Net-Net SBC with a TCP port range for:

- Q.931—Q.931 ports that are frontend ports handled by a real backend socket, and are therefore “virtual”
- Separate H.245 TCP connections—Actual sockets that the Net-Net SBC handles separately

Virtual call signaling address is an H.323 call signaling address that is registered with a gatekeeper, but does not have a corresponding listening socket in the Net-Net SBC. Using the virtual call signaling address means that numerous network transport addresses do not need to be allocated.

Virtual call signaling addresses work by attaching a range of TCP server ports to a single listening TCP socket. After a connection is accepted, the accepting socket created by the server socket operated normally, as though it were created by the server socket that listens on the same transport address as the destination of the arriving packet.

To use virtual call signaling addresses, you specify a Q.931 port range from which the Net-Net SBC can allocate ports. This port range is associated with the virtual call signal IP address you specify. To bind dynamic TCP connections to a port within a port range, you configure a dynamic H.245 port range. The dynamic H.245 port range refers to the separate TCP connection for H.245 that takes place when tunneling is not being used. This enables the Net-Net SBC to select the port to which the TCP socket is bound. These two port ranges cannot overlap.

When a new RRQ has to be forwarded to the gatekeeper, the Net-Net SBC caches the registration and then forwards a modified copy of the RRQ. The Net-Net SBC allocates a virtual call signal address on the gateway stack and uses it to replace the CSA of the registering endpoint in the forwarded RRQ.

Virtual RAS Address

The Net-Net SBC also allocates a virtual RAS address for each endpoint registration. Before forwarding an RRQ from an endpoint, the Net-Net SBC replaces the RAS address of the registering endpoint with the virtual RAS address on the gateway interface.

RAS Message Proxy

When the Net-Net SBC's registration proxy feature is configured, RAS messages to and from endpoints are forwarded, except for the following: GRQ, GCF, GRJ, IRQ, IRR, IACK, and INACK. If the Net-Net SBC receives a valid GRQ on the RAS port of the gatekeeper stack that supports H.323 registration, it responds with a GCF message. Otherwise, it sends a GRJ message.

If the gateway interface receives IRR or IRQ messages, the Net-Net SBC attempts to respond based on the information about the call, and does not forward the messages.

Other RAS messages are forwarded after some modifications:

- Translating the transport address
- Deleting fields that the Net-Net SBC does not support

For further information, about how the Net-Net SBC modifies or deletes RAS message fields, refer to the About RAS Message Treatment section at the end of this chapter.

About Setting Port Ranges

When you configure the H.323 registration proxy feature, you set the Q.931 port range and the dynamic H.245 port range for H.245 connections. If you configure a Q.931 port range, you must also configure a dynamic H.245 port range.

These port ranges cannot overlap because TCP ports must be unique. The dynamic H.245 port range is used to allocate a real TCP socket, but the Q.931 port range allocates a virtual call signaling address that does not have an associated listening TCP socket.

Note: You should choose these sockets with future Net-Net SBC features about security in mind because future development will support performing admission control based on these port ranges. You will be able to set up filtering rules to allow only inbound packets to configured port ranges.

The following table shows how the Q.931 and dynamic H.245 port ranges work. If you set the start port of 1024 and the number of ports to 1024, you will have configured a port range that starts at 1024 and ends at 2047. So the final port in the range is the start port number added to the number of points, minus 1. Remember that you cannot overlap the Q.931 and dynamic H.245 port ranges. Notice that the higher the number of the start ports, the fewer ranges of ports you have remaining from which to choose.

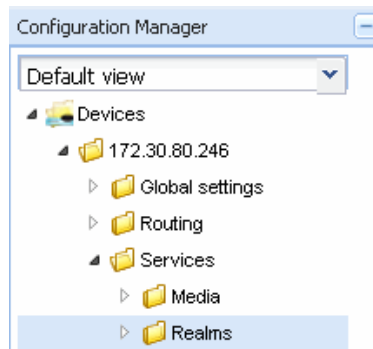
Number of Ports	Start Port	n
1024	1024 * n	1-63
2048	2048 * n	1-31
4096	4096 * n	1-15
8192	8192 * n	1-7
16384	16384 * n	1-3
32768	32768 * n	1

Configuring H.323 Registration Proxy

You enter Q.931 and dynamic H.245 port ranges to configure H.323 registration proxy.

To configure H.323 registration proxy:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Scroll to the H.323 registration proxy section.

7. **Ports in Q.931 port range (#)**—Select the number of ports to be included in the Q.931 port range to use for the call signalling address forwarded in the RRQ from the combination box.

Ports in Q.931 port range

0
4096
0
8192
32768
16384
1024
2048

Note: If you have enabled process registration for this H.323 interface, this value must be set to zero because the interface is a gatekeeper that does not support the virtual call signaling address feature.

8. **Q.931 start port**—Select the number where you want the Q.931 port range to start from the combination box. To prevent using a well-known reserved port, this number must be greater than or equal to 2048. You must change this value from the default (0).

Q.931 start port

2048

9. **Ports in TCP connection port range (#)**—Select the number of ports to be included in the port range to use for dynamic TCP connections from the combination box.

Ports in TCP connection port range

1024
4096
0
8192
32768
16384
1024
2048

10. **Dynamic start port**—Enter the starting port number of the dynamic H.245 port range.

Dynamic start port

1024

11. Click **Apply**.

H.323 Registration Caching

The Net-Net SBC can cache and proxy an H.225 RRQ between an H.323 endpoint and a gatekeeper. Registration caching has two benefits:

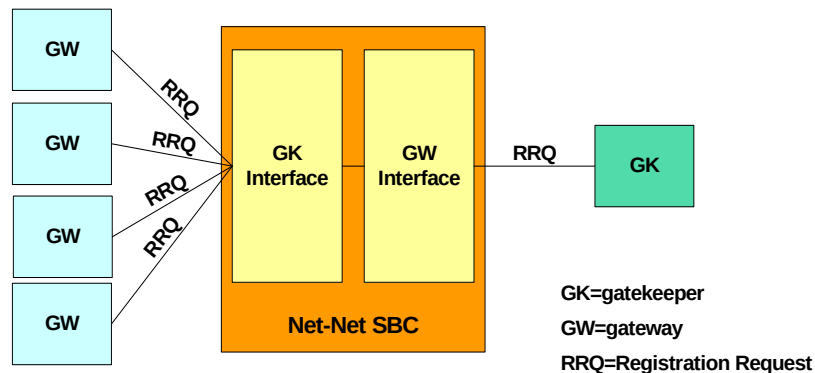
- It allows the aggregation of RRQs sent to a gatekeeper stack and proxies those requests through the gateway stack. If the external gatekeeper associated with the outbound (gateway) interface does not support additive registration, then the Net-Net SBC consolidates the requests by placing them all in the same packet. Otherwise, additive registration is used on the outbound (gateway) interface.
- It allows the gatekeeper stack to use the registration information to route calls from other realms to the endpoints in its realm.

For registration caching, you need to configure at least two H.323 interfaces:

- One gatekeeper interface to receive registrations
- One gateway interface to proxy registrations

The Net-Net SBC caches all successful registrations, using the cache to route calls back to the associated endpoint.

The following diagram shows how RRQs flow during registration caching.



Caveats for Registration Caching

This feature has the following caveats:

- If a gateway stack receives a URQ message from the gatekeeper, it confirms the request with an UCF message. It flushes all registration caching for that stack. However, the Net-Net SBC does not send URQs to the registered endpoints.
- The Net-Net SBC must be rebooted so that the gateway interface can rediscover the gatekeeper

Automatic gateway discovery is turned on for the gateway interface by setting the automatic gateway discovery parameter to **enabled**.

Configuration Requirements

For the Net-Net SBC to determine where to route an RRQ, either the associated stack parameter or the gatekeeper identifier field is used.

First, the Net-Net SBC uses the associated interface (assoc-stack) of the gatekeeper interface to find the interface for the outgoing RRQ. If you do not configure an associated interface and the incoming RRQ has a `gatekeeperIdentifier` field, the Net-Net SBC finds a configured gateway interface with a matching `gk-identifier` field and use it as the outgoing interface. If the incoming RRQ does not have a `gatekeeperIdentifier` field and the gatekeeper interface has a configured gatekeeper identifier, the Net-Net SBC finds a gateway interface with a gatekeeper identifier that matches the one set for the gatekeeper interface and then use it as the outgoing interface. If an outgoing interface cannot be determined, the Net-Net SBC rejects the RRQ with the reason `discoveryRequired`.

A configured H.323 interface can be the gateway interface for more than one gatekeeper interface. If a call is received on the gateway interface, the registration cache will be queried to find a registration matching the call's destination. If a registration is found, the interface on which the registration was received will be used as the outgoing interface for the call.

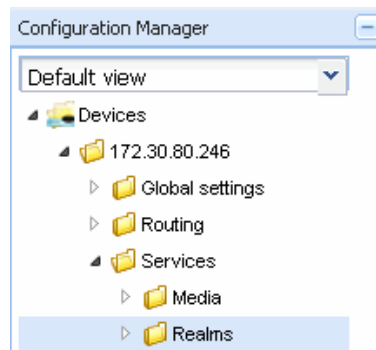
Subsequent ARQ or URQ messages coming from a registered endpoint will be proxied to the gatekeeper using the outgoing gateway interface established during registration. If a registration is not found, an ARJ or a URJ will be sent to the endpoint originating the ARQ or URQ.

A gatekeeper interface can respond to a GRQ if the GRQ is received on its RAS interface. The Net-Net SBC supports GRQ on a multicast address.

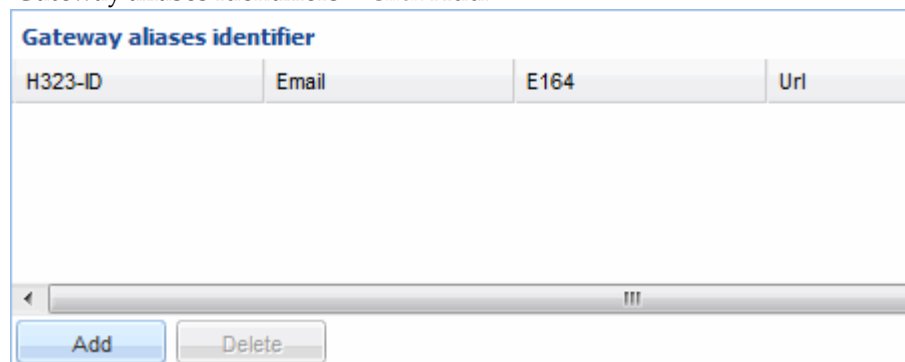
Configuring the Gateway Interface for Registration Caching

To configure the gateway interface for registration caching:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab at the top of the content area.
7. **Service mode**—Select enabled from the combination box.
8. **Gateway aliases identifiers**—Click **Add**.



The H.323 gateway alias/gatekeeper ID dialog box appears. The value you enter here is either the gateway alias or the gatekeeper identifier, depending on the

mode you configure for the interface. The aliases are set in the **sourceInfo** information element of outgoing ARQs. You must choose one field to use as an alias.

9. **Add Gateway aliases identifier**—Choose one alias type from the fields to use as an alias. Provided below are examples for each field.

Add Gateway aliases identifier
✕

H323-ID	xyz
Email	user@acmepacket.com
E164	17817566800
Url	http://acmepacket.com
IP address	63.67.143.4:2000

10. Click **Apply**. The new gateway alias identifier appears in the list.
11. **Registration expiration time-to-live (sec)**—Enter the same value as that of the `timeToLive` field in the RRQ sent to the gatekeeper.

Registration expiration time-to-live (sec)

When the Net-Net SBC receives an RCF from the gatekeeper, it extracts the `timeToLive` field and uses that value as the time interval for keeping the registration of the gateway interface alive. The Net-Net SBC sends a keep-alive RRQ about ten seconds before the registration expires.

The registration expiration you set value should not be too low because some gatekeepers simply accept the `timeToLive` in the RRQ, resulting in a potentially high volume of RRQs.

12. Scroll to Remote gatekeeper.
13. **Auto gatekeeper discovery**—Select enabled from the combination box.

Auto gatekeeper discovery disabled ▼

14. Click **Apply**.

H.245 Stage

The Net-Net SBC allows you to set the earliest stage in an H.323 call when the Net-Net SBC initiates the procedure to establish an H.245 channel for the call. If you have enabled H.245 tunneling by setting the `h245-tunneling` parameter to **enabled**, then you do not need to configure your system for this feature.

The Net-Net SBC initiates the H.245 procedure by either:

- Sending its H.245 address, or

- Creating a TCP connection to an H.245 address that it has received

You can set this parameter to any of the following stages of an H.323 call: **setup**, **proceeding**, **alerting**, **connect**, **early**, **facility**, **noh245**, and **dynamic**. With the exception of **early**, **noh245**, and **dynamic**, these values correspond to types of H.225/Q.931 messages. The **dynamic** value is described in detail in the next section.

When you configure the **early** value, your Net-Net SBC begins the H.245 procedure at the time the Setup message is sent or received, or when the Connect message is received.

While these values allow for some flexibility about when the H.245 process is started, they are inherently static. All calls in the H.323 stack configuration use the same value, and it cannot be changed from call to call on that stack.

Dynamic H.245 Stage Support

You can configure your Net-Net SBC for dynamic H.245 support, meaning that the point at which the H.245 process begins can be determined dynamically. To support dynamic H.245, the Net-Net SBC sends its H.245 address in the incoming call when it receives an H.245 address in the outgoing call.

Dynamic H.245 Stage for Incoming Calls

When a call comes in on an H.323 interface that you have configured for dynamic H.245 stage support.

The Net-Net SBC includes its H.245 address in the h245Address field of the first H.225/Q.931 message. The Net-Net SBC does this after it receives the first H.225/Q.931 message with an H.245 address in the outgoing call. Based on the first H.225/Q.931 message received by the Net-Net SBC that has an H.245 address, the Net-Net SBC selects the message in which to include the H.245 address as outlined in the table.

Message Received with H.245 Address	Message Sent with H.245 Address
Call Proceeding	Call Proceeding, Progress, Alerting, Connect or Facility. The H.245 address is sent in the Call Proceeding message if the Net-Net SBC has not sent a Call Proceeding message in the incoming call. This is true only when you enable the Fast Start in first message parameter for the incoming stack; this parameter establishes whether or not Fast Start information must be sent in the first response to a Setup message. Otherwise, the message in which the H.245 address is sent depends on what message is received after the Call Proceeding message. This is because the Net-Net SBC sends its Call Proceeding message directly after receiving the Setup message.
Progress	Progress
Alerting	Alerting
Connect	Connect
Facility	Facility

When it receives the first H.225/Q.931 message with an H.245 address in the outgoing call, the Net-Net SBC creates a listening socket on the incoming interface. It also includes the socket address and port in the H.245 address of the next H.225/Q.931 message that it sends. If there is no pending H.225/Q.931 message for the Net-Net SBC to send, it instead sends a Facility message with the reason startH245. Then the H.245 channel is established when a TCP connection is made to the listening socket.

For the outgoing leg of a call that came in on the H.323 stack configured for H.245 dynamic stage support, the Net-Net SBC starts establishing the H.245 channel when it receives the first H.225/Q.931 message with H.245 address information. It also starts to establish a TCP connection to the address and port specified in the H.245 address information. The H.245 channel for the outgoing call is established while the H.245 address (h245Address) is sent in the incoming call as described above.

Dynamic H.245 Stage for Outgoing Calls

This section describes what happens when a message exits the Net-Net SBC on an H.323 stack that you have configured for dynamic H.245 stage support.

When the Net-Net SBC receives the first H.225/Q.931 message that has H.245 address information, it establishes an H.245 channel. The Net-Net SBC initiates a TCP connection to the address and port specified in the H.245 address information.

If the incoming call for the session is also on an H.323 stack with dynamic H.245 configured, the Net-Net SBC starts the H.245 procedure in the incoming call.

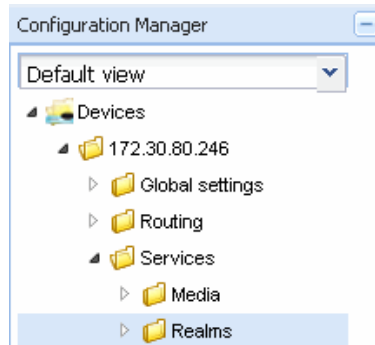
Otherwise, the Net-Net SBC sends its H.245 address in the incoming call based on the H.245 stage support that you have configured.

The process is different when the Net-Net SBC receives a TCS message on the outgoing call before the incoming call reaches its H.245 stage. In this instance, the Net-Net SBC sends a Facility message with the reason startH245 with its H.245 address in order to start the H.245 procedure. The reason is needed in order for the Net-Net SBC to exchange TCS messages with the incoming side of the call.

Configuring H.245 Stage Support

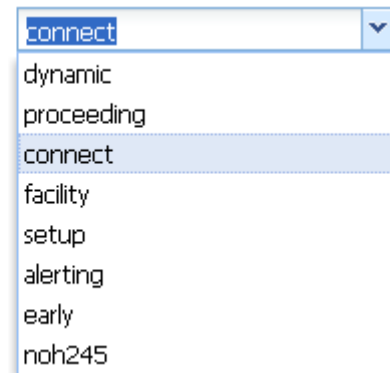
To configure H.245 stage support:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab at the top of the content area.
7. **H.225 call stage to initiate H.245 procedures**—Choose the stage at which the Net-Net SBC transfers the H.245 address to the remote side of the call, or acts on the H.245 address sent by the remote side.

H.225 call stage to initiate H.245 procedures



8. Click **Apply**.

H.323 HNT for Outbound Sessions

This section explains how H.323 hosted NAT traversal (HNT) for outbound sessions works and how to enable this capability on your Net-Net SBC.

The feature enables endpoints behind NATs to originate calls by resolving the address differences between the NAT and the actual endpoint.

H.323 communication through a NAT becomes an issue when engaging in RAS messaging. While the H.323 standard specifies specific information elements in the RAS messages that indicate the address to which the replies should be sent, these addresses will be behind the NAT and therefore unroutable. The Net-Net SBC solves this problem by sending RAS replies to the layer 3 address from which the associated RAS request was received.

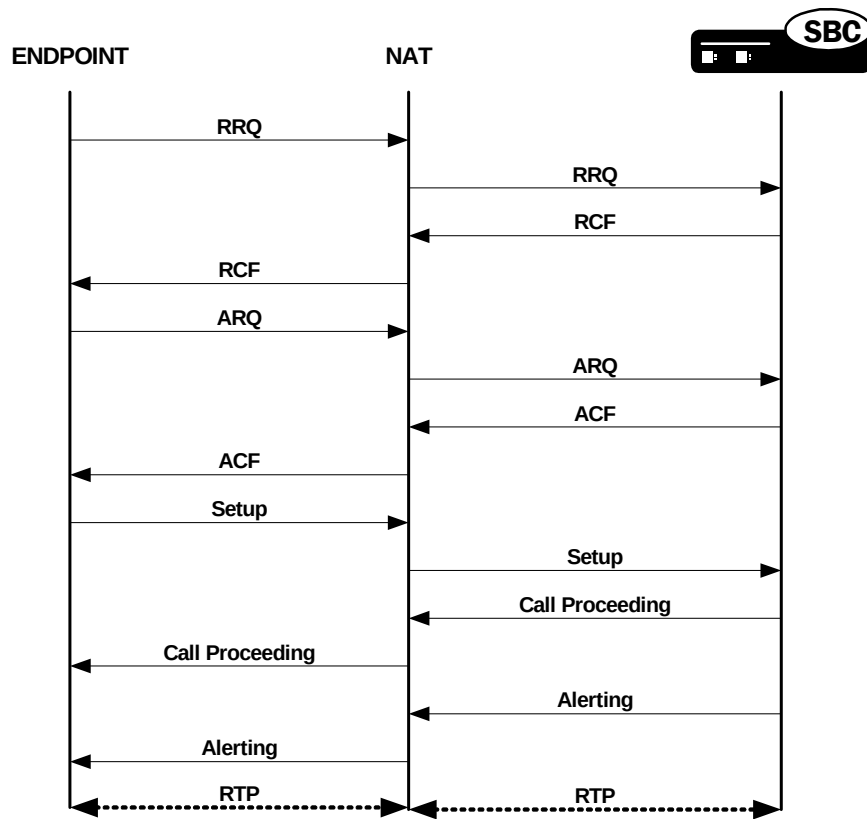
A second issue exists for media channels as the address specified in the H.323 OLC message will be behind the NAT and likewise unroutable. This is resolved by relying on the fact that the forward and reverse channels will utilize the same address and port on the endpoint. By sending media packets to the same address from which the packet are received, media and flow through the NAT.

If you do not use H.323 HNT for outbound sessions, the following behavior will occur:

- When an H.323 endpoint is behind a NAT and it registers with a gatekeeper through the Net-Net SBC, the Net-Net SBC tries to send a response back to the endpoint's RAS address rather than to the NAT from which the request was received.
- The same is true for LRQ and IRQ messages because responses without H.323 HNT for outbound sessions, responses were being sent back to the replyAddress or the rasAddress.
- In addition, the Net-Net SBC always induces one-way media because it tries to send the RTP to the media IP address and port it receives in the OLC messages rather than the ephemeral port on the intermediary NAT.

With this ability enabled, however, the Net-Net SBC sends RAS responses back to the address from which the request was received (the NAT). It does not send responses to the endpoint's rasAddress or replyAddress mentioned in the signaling message. The same is true for RTP. With H.323 HNT for outbound sessions enabled, the Net-Net SBC sends RTP to the IP address and port from which it receives the RTP packets (the NAT).

The call flow below illustrates how this feature works:



Caveats

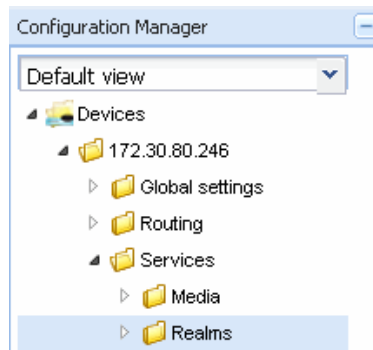
Keep in mind the following caveats when you are enabling H.323 HNT for outbound sessions on your Net-Net SBC:

- The Net-Net SBC cannot initiate sessions with H.323 endpoints that are behind a NAT. This type of session will always initiate with an H.323 endpoint, flow through the NAT, and then to the Net-Net SBC.
- This capability does not apply to calls that require IWF translation between SIP and H.323.

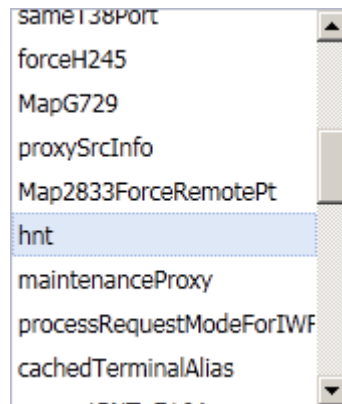
Configuring H.323 HNT for Outbound Sessions

To configure H.323 HNT for outbound sessions:

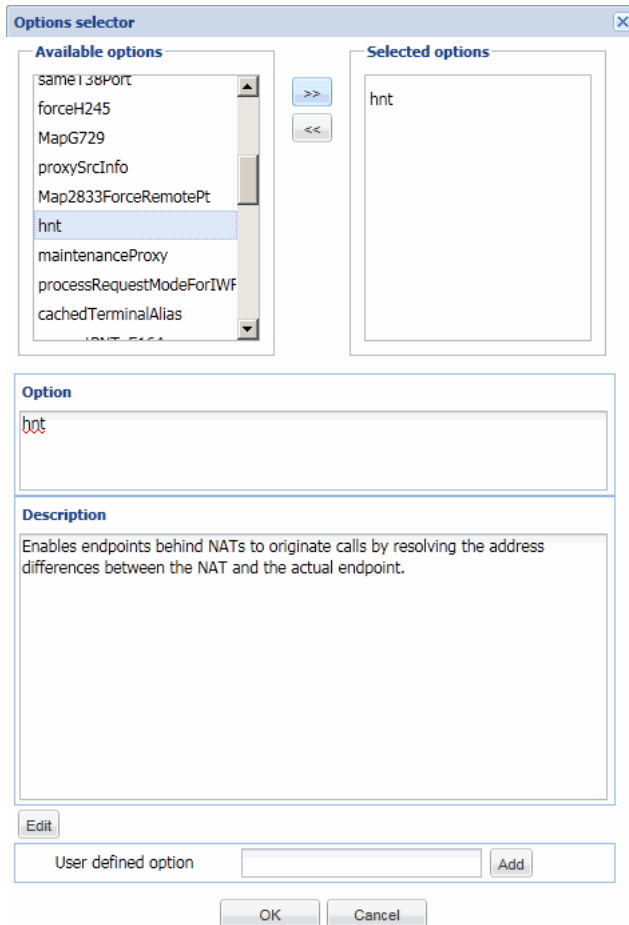
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab.
7. Scroll to the Options for H323 Service section and click **Options**. The Options dialog box appears.
8. Click **hnt** in the Available options list to select it.



9. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It contains two lists: **Available options** and **Selected options**. The **Available options** list includes: same138Port, forceH245, MapG729, proxySrcInfo, Map2833ForceRemotePt, **hnt** (highlighted), maintenanceProxy, processRequestModeForIWF, and cachedTerminalAlias. The **Selected options** list contains: hnt. Below the lists, the **Option** field displays **hnt** and the **Description** field displays: "Enables endpoints behind NATs to originate calls by resolving the address differences between the NAT and the actual endpoint." At the bottom, there is an **Edit** button, a **User defined option** field with an **Add** button, and **OK** and **Cancel** buttons.

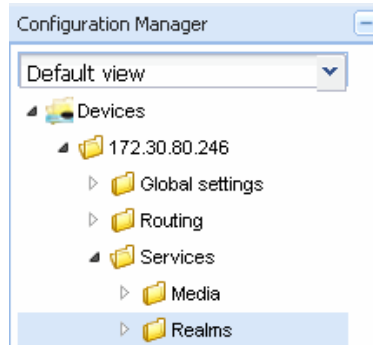
10. Click **OK**.

H.323 Party Number-E.164 Support

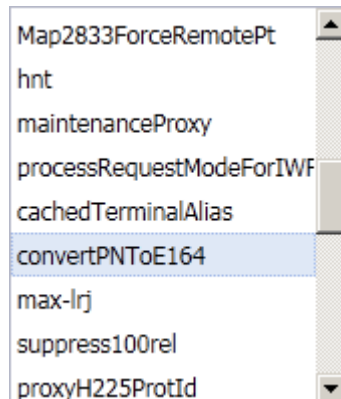
Some H.323 gateways cannot handle **partyNumber** alias addresses in H.225 messages. The Net-Net SBC lets you convert this address type to **dialedDigits** (E.164). This conversion applies to **sourceAddress**, **destinationAddress**, and **destExtraCallInfo** aliases in Setup messages.

To configure H.323 party number-E.164 support:

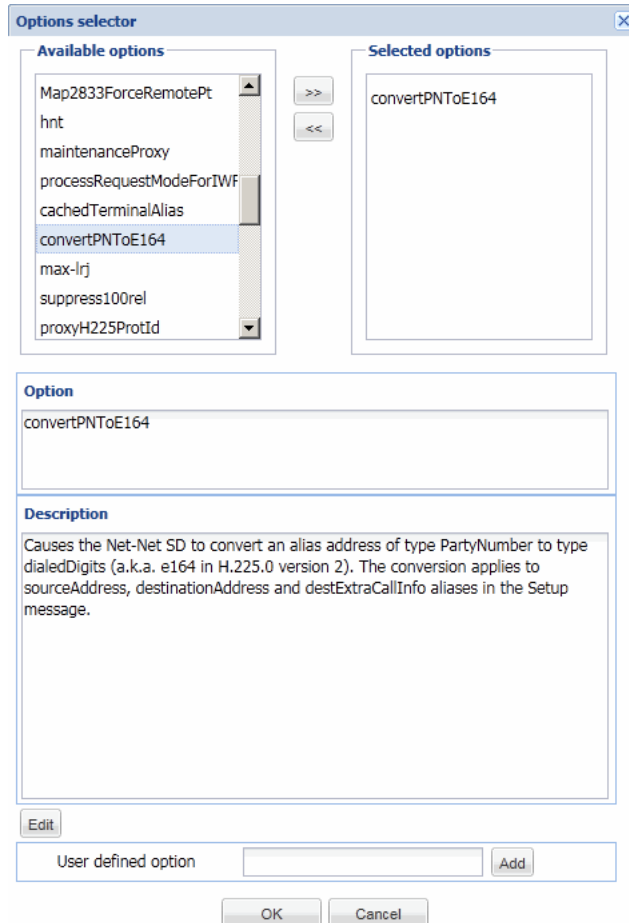
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab.
7. Scroll to the Options for H323 Service section and click **Options**. The Options dialog box appears.
8. Click **convertPNTtoE164** in the Available options list to select it.



9. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It contains two main sections: **Available options** and **Selected options**. In the **Available options** list, **convertPNTToE164** is selected. The **Selected options** list is currently empty. Below these lists are two buttons: **>>** and **<<**. The **>>** button is highlighted. Below the lists, there is a section for the selected option, showing the **Option** name **convertPNTToE164** and its **Description**: "Causes the Net-Net SD to convert an alias address of type PartyNumber to type dialedDigits (a.k.a. e164 in H.225.0 version 2). The conversion applies to sourceAddress, destinationAddress and destExtraCallInfo aliases in the Setup message." At the bottom, there is an **Edit** button, a text field labeled **User defined option** with an **Add** button next to it, and **OK** and **Cancel** buttons.

10. Click **OK**.

Signaling Only Operation

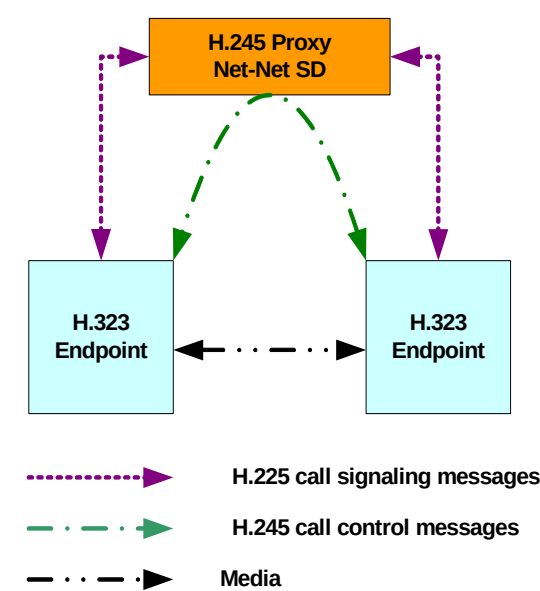
When you set the Net-Net SBC to operate in signaling-only mode, it acts like a signaling server. It proxies the call signaling messages between two endpoints. Note, however, that the Net-Net SBC does not function as a RAS proxy; it does not proxy RAS messages.

You have two options for the proxy mode:

- H.245 proxy mode—The Net-Net SBC handles call signaling (H.225) and call control (H.245) messages.
- H.225 proxy mode—The Net-Net SBC handles call signaling

H.245

When in H.245 proxy mode, the Net-Net SBC proxies or passes through the call signaling (H.225) messages and the call control (H.245) messages. It allows media to flow between the two H.323 endpoints, as shown in the following diagram.



In some deployments, the media might be treated by a NAT device. When the Net-Net SBC is in H.245 proxy mode, any tunneled H.245 message on the ingress side is tunneled in the egress side. However, if the tunneling is refused on the egress side, a separate H.245 session is established.

H.245 proxy mode support is defined in the following table.

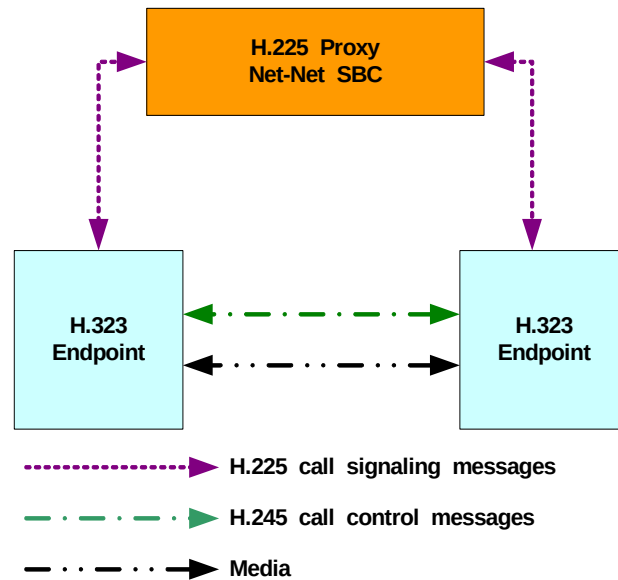
Ingress	Egress
Tunneled	Tunneled
Tunneled	Separate H.245 session
Separate H.245 session	Tunneled
Separate H.245 session	Separate H.245 session

H.225

When in H.225 proxy mode, the Net-Net SBC only proxies call signaling (H.225 messages). The call control (H.245 messages) and the media associated with the session do not go through the Net-Net SBC. Instead, they flow directly between the two H.323 endpoints.

Note: H.225 proxy mode is only used in specific applications and should not be enabled without consultation from your Acme Packet Systems Engineer.

The following diagram shows the flow.



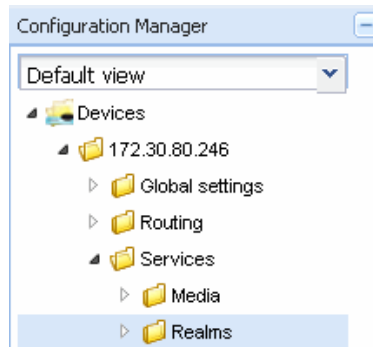
In certain deployments, the call control message and media are exchanged between the two H.323 endpoints themselves. When the Net-Net SBC is in H.225 proxy mode, any tunneled H.245 message on the ingress side is tunneled in the egress side; this is irrespective of the value configured for H.245 tunneling in the H.323 interface configuration.

Configuring H.323 Signaling Only Operation

To use this feature, you need to set the proxy mode parameter in the H.323 interface configuration.

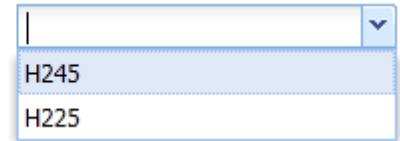
To configure H.323 signaling only operation:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab.
7. **Signaling proxy mode (no media)**—Choose the proxy mode from the combination box.

Signaling proxy mode (no media)



8. Click **Apply**.

Maintenance Proxy Function

The Net-Net SBC supports a maintenance proxy function for H.323 and enhances the way the Net-Net SBC creates unique RAS ports. You can register endpoints through the Net-Net SBC with unique RAS port. You can also set the H.323 interface on the enterprise side to represent enterprise-side endpoints and thereby register on the carrier side.

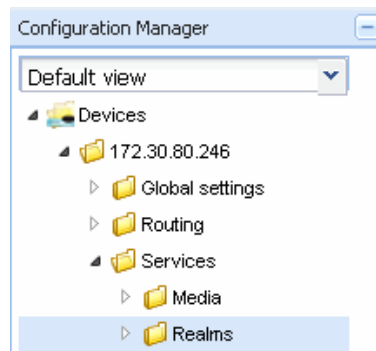
The maintenance proxy creates a many-to-one association between the enterprise and the carrier side. Interfaces on the enterprise side can be associated with the carrier side interface, which also must be configured to for the maintenance proxy feature.

Configuring the Maintenance Proxy Function

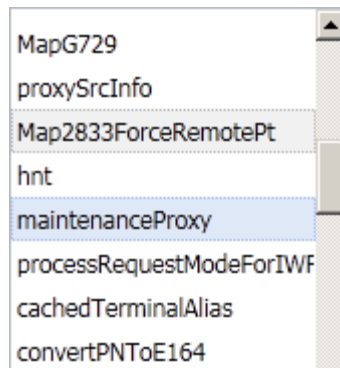
To configure the maintenance proxy function:

You need to set two values in the options parameters for the H.323 interface.

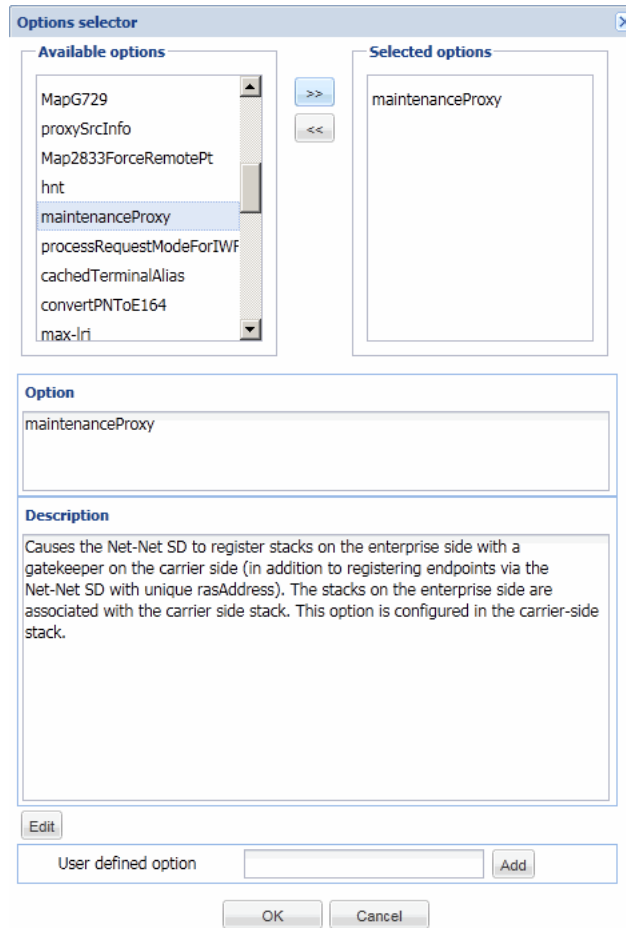
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab.
7. Scroll to Options for H323 Service and click **Options**. The Options dialog box appears.
8. Click **maintenanceProxy** in the Available options list to select it.



9. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** list contains: MapG729, proxySrcInfo, Map2833ForceRemotePt, hnt, maintenanceProxy (highlighted), processRequestModeForIWF, cachedTerminalAlias, convertPNTToE164, and max-lri. Between the lists are two buttons: a right arrow (>>) and a left arrow (<<). The **Selected options** list contains: maintenanceProxy. Below these lists, there is a section for the selected option. It has a label **Option** with the value 'maintenanceProxy' in a text field. Below that is a label **Description** with a text area containing the following text: 'Causes the Net-Net SD to register stacks on the enterprise side with a gatekeeper on the carrier side (in addition to registering endpoints via the Net-Net SD with unique rasAddress). The stacks on the enterprise side are associated with the carrier side stack. This option is configured in the carrier-side stack.' At the bottom left of this section is an **Edit** button. Below the description is a section for **User defined option** with a text input field and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

10. Click **OK**.
11. Click **Apply**.

TCP Keepalive

You can configure the Net-Net SBC to control TCP connections by setting:

- The amount of time the TCP connection is idle before the Net-Net SBC starts sending keepalive messages to the remote peer
- The number of keepalive packets the Net-Net SBC sends before terminating the TCP connection

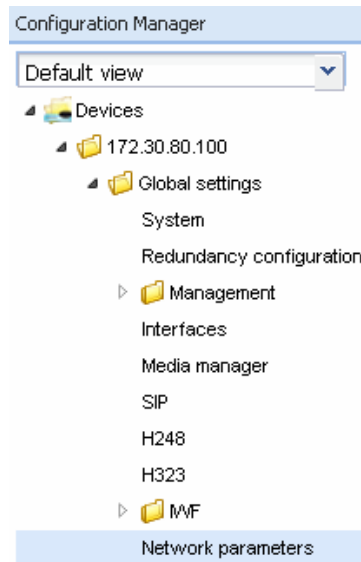
If TCP keepalive fails, then the Net-Net SBC will drop the call associated with that TCP connection. TCP settings are global, and then enabled or disabled on a per-interface basis.

Configuring Global TCP Keepalive

To configure TCP keepalive parameters on your Net-Net SBC:

If you want to use the default values for TCP keepalive, you can skip the global configuration and just set the TCP keepalive function in the H.323 interface configuration. The global network defaults will be applied.

1. Expand the Configuration Manager slider > Global settings.



2. Click Network parameters.
3. Press **Add** to add the configuration in the content area.
4. **Keepalive timer (sec)**—Enter the number of seconds is idle time before TCP keepalive messages are sent to the remote peer.

Keepalive timer(sec)

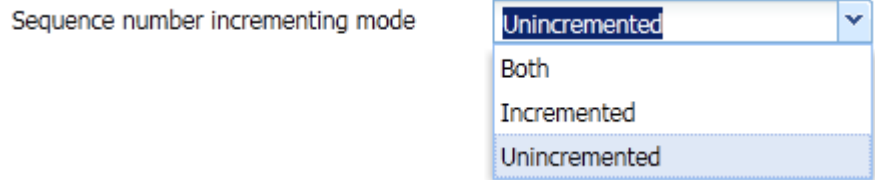
7200

5. **Max # of TCP keepalive retries**—Enter the number of packets the Net-Net SBC sends to the remote peer before it terminates the TCP connection.

Max number of TCP keepalive retries

8

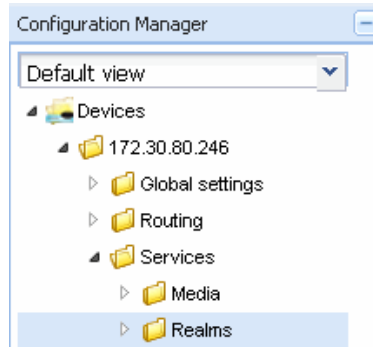
6. **Sequence number incrementing mode**—Choose the mode from the combination box.



Applying TCP Keepalive to the H.323 Interface

To apply these settings individually per H.323 interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 interface tab.
7. **TCP keepalives**— Select enabled from the combination box and click it to enable TCP keepalives.



8. Click **Apply**.

Automatic Gatekeeper Discovery

Available only when the H.323 interface is functioning as a gateway, this feature allows for automatic gatekeeper discovery on start-up.

This feature is based on the Net-Net SBC sending a GRQ to the multicast address of the RAS Multicast Group, which is the device group listening on this address. If you do not configure a multicast address, Net-Net SBC uses the well-known address and port 224.0.1.41:1718 in the address-port combination making up this parameter.

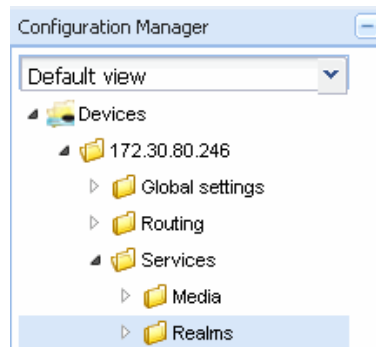
Multicast only functions when the Net-Net SBC is discovering an external gatekeeper. The Net-Net SBC does not respond to multicast gatekeeper queries.

When it receives a GCF message from a gatekeeper, the Net-Net SBC registers with the gatekeeper indicated in the GCF. When it receives an GRJ message that contains optional information about alternative gatekeepers, the Net-Net SBC attempts to register with an alternate.

If you do not use automatic gatekeeper discovery, the Net-Net SBC registers with the gatekeeper you configure in the gatekeeper parameter. In this case, the gatekeeper identifier you configure is included in to the RRQ. No registration takes place if you do not establish automatic gatekeeper discovery or if you do not configure the gatekeeper and its identifier.

To configure automatic gatekeeper discovery:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab at the top of the content area.
7. Scroll to the Remote gatekeeper section.
8. **Auto gatekeeper discovery**—Choose auto-discovery from the combination box.



9. **RAS multicast group IP address:port**—Enter the address and port where the RAS Multicast Group listens. If you do not set this parameter, then the Net-Net SBC will use the well-known address and port 240.0.1.41:1718. Your entries in this field will be comprised of an IP address and port value.

RAS multicast group IP address:port

ipAddress=240.0.1.41:1718

10. Click **Apply**.

H.323 Alternate Routing

You can configure your Net-Net SBC to try more possible routes within given time constraints and number of retries.

Without Alternate Routing Enabled

If you do not enable H.323 alternate routing, the Net-Net SBC tries one possible next hop gateway when routing H.323 calls even if the applicable local policy has multiple next hops configured. If that next hop gateway fails (either because it is busy or out of service), the Net-Net SBC relays the failure back to the caller, who hears a busy tone.

In addition, the call will only be routed to the other available next hops if the first one is:

- A session agent that has gone out of service because its constraints have been exceeded
- A session agent that has gone out of service because it failed to respond to a Net-Net SBC Setup request
- A session agent group

With Alternate Routing Enabled

When you enable H.323 alternate routing on your Net-Net SBC, you enable the use of the other next hops in addition to the first one. The retry, when the other available next hops are used, is transparent to the caller. However, the number of retries is limited by the value you set for the Call establishment-SETUP to CONNECT message interval parameter, and this feature works only if there is more than one matching local policy next hop. If there is not more than one match, even if that match is a session agent group, then the call is only attempted once and the caller must retry it.

If the Net-Net SBC receives a Release Complete message before it receives an Alerting message, then it will try the next hop if there are multiple matches. When there is only one match, or if the timer or number of retries is exceeded, the Net-Net SBC proxies the most recently received Release Complete message back to the caller.

The following table shows the cause codes and release complete reasons, and either of the two actions the Net-Net SBC takes:

- **Recur**—Means that the Net-Net SBC performs (or continues to perform) alternate routing

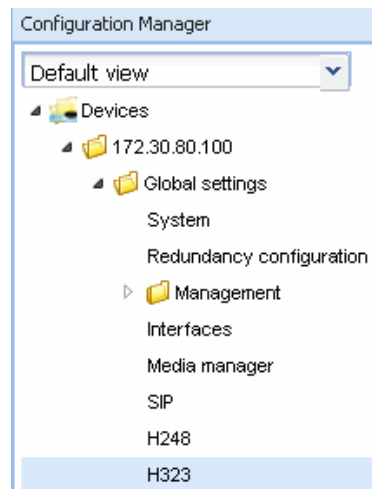
- Proxy—Means that alternate routing stops, and the Net-Net SBC sends a release complete message back to the caller

H.323 Release Complete Reason	Q.850 Cause Code	Action
No Bandwidth	34—No circuit available	Recur
Gatekeeper Resources	47—Resource unavailable	Recur
Unreachable Destination	3—No route to destination	Recur
Destination Rejection	16—Normal call clearing	Proxy
Invalid Revision	88—Incompatible destination	Recur
No Permission	111—Interworking, unspecified	Recur
Unreachable Gatekeeper	38—Network out of order	Recur
Gateway Resources	42—Switching equipment congestion	Recur
Bad Format Address	28—Invalid number format	Recur
Adaptive Busy	41—Temporary Failure	Recur
In Conference	17—User busy	Proxy
Undefined Reason	31—Normal, unspecified	Recur
Facility Call Deflection	16—Normal, call clearing	Proxy
Security Denied	31—Normal, unspecified	Recur
Called Party Not Registered	20—Subscriber absent	Recur
Caller Not Registered	31—Normal, unspecified	Recur
New Connection Needed	47—Resource Unavailable	Recur
Non Standard Reason	127—Interworking, unspecified	Recur
Replace With Conference Invite	31—Normal, unspecified	Recur
Generic Data Reason	31—Normal, unspecified	Recur
Needed Feature Not Supported	31—Normal, unspecified	Recur
Tunnelled Signaling Rejected	127—Interworking, unspecified	Recur

Configuring H.323 Alternate Routing

To enable H.323 alternate routing:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. **Response to SETUP message sent**—Enter the time in seconds for the response time-out (or T303 timer). This is the amount of time allowed to elapse during which the Net-Net SBC should receive a response to its Setup message. If the first response to the Net-Net SBC's Setup is a callProceeding, then the Net-Net SBC should receive an Alerting or Connect message before this timer (now T303*2) elapses.

Response to SETUP message sent

4

5. **Call establishment-SETUP to CONNECT message interval**—Enter the time in seconds for the connect time-out (or T301 timer). This is the amount of time allowed to elapse during which the Net-Net SBC should receive a Connect message.

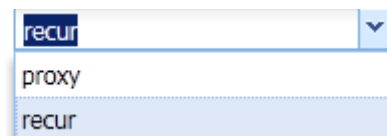
For alternate routing, this parameter is also used to limit the number of next hops that are tried and the length of time they are tried in case the first next hop fails. The call needs to be established before this timer expires; the call will fail after maximum of 5 retries.

Call establishment-SETUP to CONNECT
message interval

32

6. **Alternate routing**—Choose **recur** from the combination box.

Alternate routing



7. Click **Apply**.

H.323 Bearer-Independent Setup

The Net-Net SBC supports an H.323 option that enables H.323 Bearer-Independent Setup (BIS). When enabled, this feature allows exception to slow-start to fast-start conversion on the Net-Net SBC.

H.323 BIS Disabled

Unless you enable this feature, the Net-Net SBC performs slow-start to fast-start conversion when a call entering the system as slow-start was routed to an outgoing H.323 interface (stack) with **call-fast-start** set to enabled and there is a list of valid media-profiles in the configuration.

H.323 BIS Enabled

There are certain cases in access deployments where the slow-start to fast-start conversion should not be applied. This is the case when the Setup message contains the Bearer Capability information element (IE), which signals BIS.

When you enable this feature and the Net-Net SBC receives an incoming Setup message that does not contain a fastStart field, the Net-Net SBC checks for the BIS in the incoming Setup before it starts to perform the slow-start to fast-start conversion. If it finds the BIS, then it does not perform the conversion.

This feature can be enabled on a global or a per-interface basis, meaning that you can apply it to your system's entire H.323 configuration or you can enable it only for the interfaces where you want it applied.

Configuring H.323 BIS

You can configure H.323 BIS using the ACLI or Net-Net Central.

ACLI Instructions and Examples

This section explains how to add H.323 BIS support to your global H.323 configuration and to specific H.323 interfaces (stacks).

If you set this option on an H.323 interface (stack), you must set it on the interface (stack) that receives the Setup message with BIS in the Bearer Capability IE.

To enable the H.323 BIS feature globally:

1. Expand the **session-router** folder and click **h323-stack**.
2. Click the H.323 stack configuration you want to configure in the table and click **Edit**.
3. Scroll to options and click **Options**. The Options selector dialog box appears.
4. Click **bearerIndSetup** in the Available options list to select it.
5. Click  to move the option to the Selected options list. A description of this option appears in the Description field.
6. Click **OK**. The option is added to the options table.
7. Click **Apply**.

To enable the H.323 BIS feature for a specific H.323 interface:

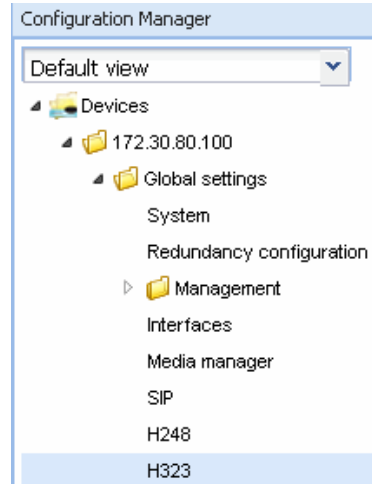
1. Expand the **session-router** folder and click **h323-stack**.
2. Click the H.323 stack configuration you want to configure in the table and click **Edit**.

3. Scroll to options and click **Options**. The Options selector dialog box appears.
4. Click **bearerIndSetup** in the Available options list to select it.
5. Click  to move the option to the Selected options list. A description of this option appears in the Description field.
6. Click **OK**. The option is added to the options table.
7. Click **Apply**.

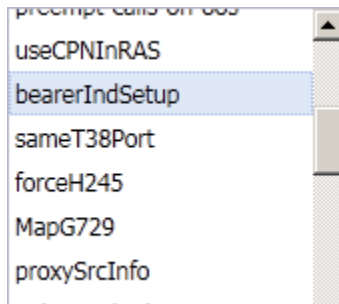
Configuring H.323 Bearer-Independent Setup

To enable the H.323 BIS feature globally:

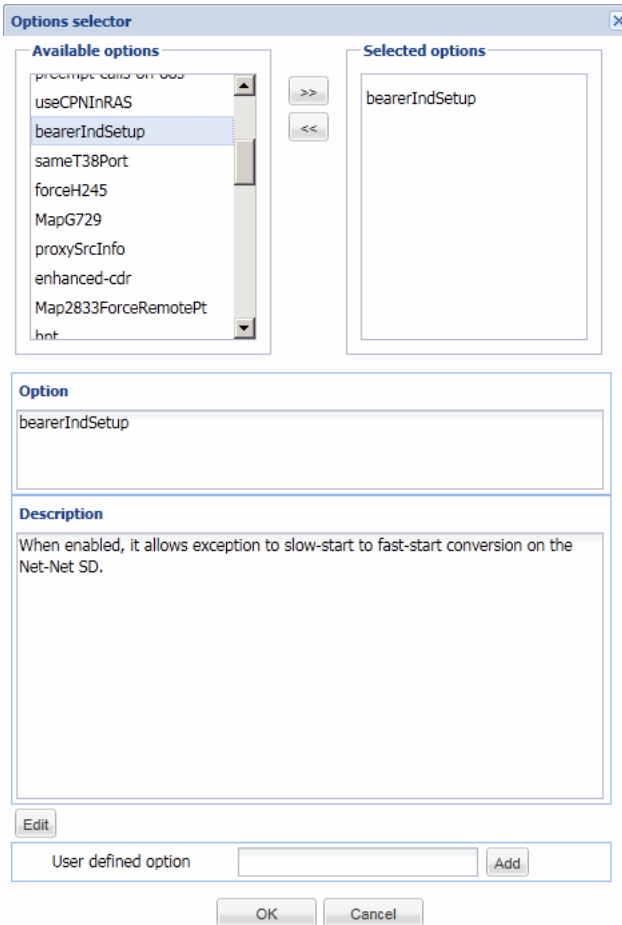
1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Click **Options**. The H.323 option dialog box appears.
5. Click **bearerIndSetup** in the Available options list.



6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.

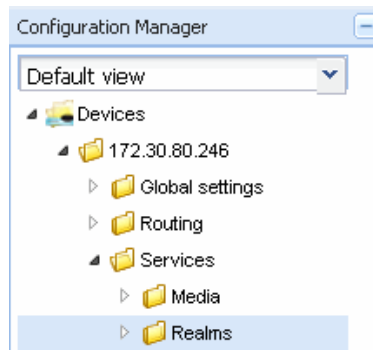


The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** section contains a list box with the following items: `preemptCallOnRas`, `useCPNInRAS`, `bearerIndSetup` (which is selected), `sameT38Port`, `forceH245`, `MapG729`, `proxySrcInfo`, `enhanced-cdr`, `Map2833ForceRemotePt`, and `hnt`. Between the two list boxes are two buttons: `>>` and `<<`. The **Selected options** section contains a list box with the item `bearerIndSetup`. Below these sections, there is a detailed view for the selected option. It has a label **Option** with the text `bearerIndSetup` in a text box. Below that is a label **Description** with a text box containing the text: "When enabled, it allows exception to slow-start to fast-start conversion on the Net-Net SD." At the bottom of the dialog, there is an **Edit** button, a text box labeled "User defined option", an **Add** button, and **OK** and **Cancel** buttons.

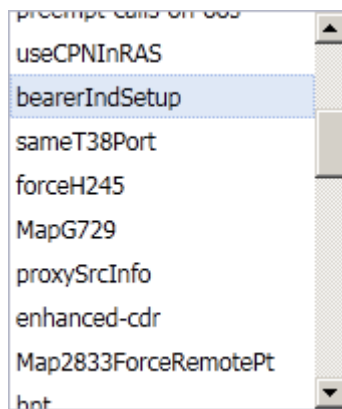
7. Click **OK**.
8. Click **Apply**.

To enable the H.323 BIS feature for a specific H.323 interface:

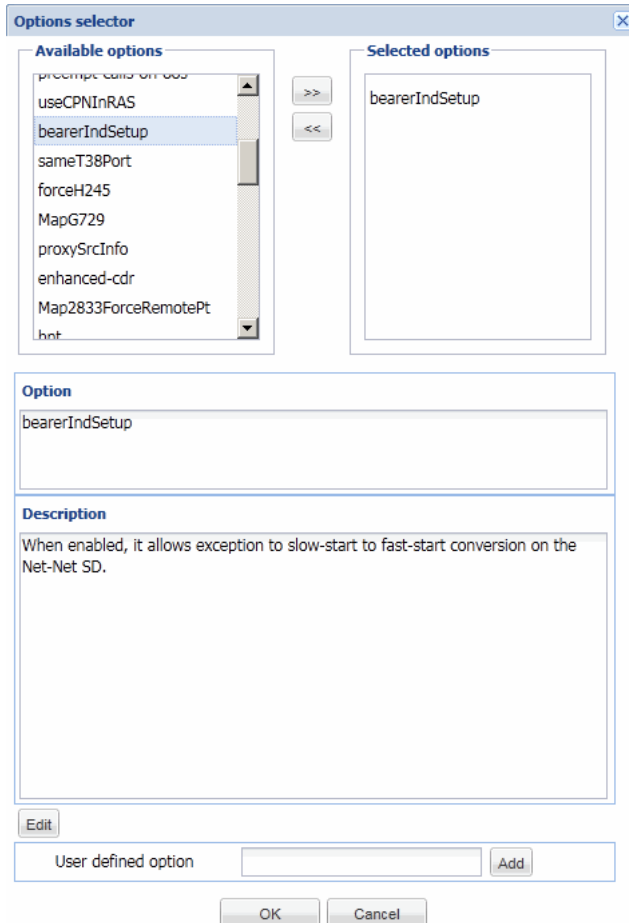
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab.
7. Click **Options**. The H.323 option dialog box appears.
8. Click **bearerIndSetup** in the Available options list.



9. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.



The image shows a screenshot of the 'Options selector' dialog box. It has a title bar with a close button. Inside, there are two main sections: 'Available options' and 'Selected options'. The 'Available options' section contains a list box with the following items: 'preempt call on qos', 'useCPNInRAS', 'bearerIndSetup' (which is highlighted), 'sameT38Port', 'forceH245', 'MapG729', 'proxySrcInfo', 'enhanced-cdr', 'Map2833ForceRemotePt', and 'hot'. Between the two list boxes are two buttons: '>>' and '<<'. The 'Selected options' section contains a list box with the item 'bearerIndSetup'. Below these sections, there is a detailed view for the selected option. It has a label 'Option' with the text 'bearerIndSetup' and a label 'Description' with the text 'When enabled, it allows exception to slow-start to fast-start conversion on the Net-Net SD.' At the bottom of the dialog, there is an 'Edit' button, a 'User defined option' text field, an 'Add' button, and 'OK' and 'Cancel' buttons.

10. Click **OK**.
11. Click **Apply**.

TOS Marking for H.323 Signaling

You can configure your Net-Net SBC to perform TOS/DiffServ marking for H.323 signaling packets. You can mark H.323 signaling packets so that they receive specific treatment from upstream devices. You can configure the TOS byte inserted in the H.323 packet to mark the traffic for certain destinations. For example, you can prevent unauthorized video transmission through an audio-only session.

The Net-Net SBC also performs TOS/DiffServ marking for media. For more information, refer to this guide's *Realms and Nested Realms* chapter. The *Realms and Nested Realms* chapter also contains more information about TOS and DiffServ in general. Refer to that chapter for configuration steps for both kinds of TOS/DiffServ marking: media and signaling.

H.323 Options

The options parameter in the global H.323 and H.323 interface configurations allows you to establish the use of specific features; most of those features are customer specific.

You should exercise caution when you apply options because of the fact that many of them are for customer-specific applications. Consult with your Acme Packet systems engineering to find out if using a particular option would be an advantage to you.

Under no circumstance do we recommend that you configure options without Acme Packet consultation. There is the chance that you could set an option that might harm an otherwise sound configuration.

Some of the options described below are only applicable to IWF calls. However, you need to establish them in your H.323 configuration.

Global H.323 Options

The following table lists the options that you might want to use in the global H.323 configuration. Again, we recommend that you consult with an Acme Packet systems engineer about your configuration before using any of these options.

Options	Description
NoDynamicMSD	Net-Net SBC forcefully assumes the “master” role for an outgoing call, and the “slave” role for an incoming call.
AllowOLCWoMSD	Net-Net SBC sends OLC before master/slave determination is complete. Causes the Net-Net SBC to be noncompliant with the H.323 recommendation, which does not permit an OLC to be sent prior to MSD completion.
ModifyMediaInAck	Net-Net SBC accepts and propagates changes to media presented in an OLC Ack. - Applies only to Fast Start OLC/OLC Ack messages embedded in H.225/Q.931 messages during call setup. - Causes Net-Net SBC to be noncompliant with the H.323 recommendation, which does not permit media characteristic to be specified in an OLC to be changed in an OLC Ack.
MapG729	Net-Net SBC maps H.245 G.729 to SDP G.729 with Annex B and vice versa. Applicable only to IWF calls.
ColonG729	Net-Net SBC uses the : (colon) instead of the = (equal sign) in the media attribute line a=fmtp:18 annexb=yes/no when mapping H.245 G.729 or SDP G.729 with Annex B. Applicable only to IWF calls.
IwfLRQ	Net-Net SBC sends an INVITE (with no SDP) to a redirect server in response to an incoming LRQ received on an H.323 interface. If a 3xx message with a redirected contact header is returned, the Net-Net SBC will send an LCF in response to the LRQ. Otherwise, it will send an LRJ.
NoG729AnnexB	SDP received by the IWF with H.729 and no FMTP will be mapped to G.729 on the H.323 side of the call. Can also be set in the session agent options parameter.
sameT38Port	Net-Net SBC does not allocate separate ports for audio and T.38. Net-Net SBC will send the same audio port in the OLC Ack that it sees in a request mode for T.38 and a new OLC for T.38.

Options	Description
pvtStats	Net-Net SBC includes program value tree (PVT) statistics in the "show h323d" display that are a sum of the PVT statistics for all H.323 interfaces. Used for debugging purposes.
strayARQTimer	Required the syntax "strayARQTimer=x," where x is the number of seconds the Net-Net SBC waits before tearing down an unsuccessful call in the case of stray ARQs.

H.323 Interface Options

The following table lists the options that you might want to use in the configuration H.323 interfaces. Again, we recommend that you consult with an Acme Packet systems engineer about your configuration before using any of these options.

Option	Description
stackAliasWins	Net-Net SBC will replace the sourceAddress of the incoming Setup message with the terminal alias of the egress interface when copying the incoming sourceAddress to the outgoing Setup message.
uniqueRRQRASAddress	Net-Net SBC will generate unique rasAddress for each RRQ that it sends to a gatekeeper in response to an incoming RRQ received on an H.323 interface configured for process registration. The IP address will be the local-ip of the outgoing interface, so the port is the unique portion of the rasAddress.
nonV4AdditiveRRQ	Gatekeeper associated with the H.323 interface support additive registration even though it does not set the additiveRegistration field in the RRQ message. When sending in the additive mode, the H.323 interface only sends with the RRQ new terminal aliases that need to be registered. In non-additive mode, the interface sense all the terminal aliases that have been registered, plus the new aliases.
cachedTerimnalAlias	Net-Net SBC copies the terminal alias(es) of the registered endpoint to the asourceAddress field of the Setup message. Terminal alias(es) are changed after the Net-Net SBC successfully processes an RRQ from the endpoint.
proxySrcInfo	Net-Net SBC copies the sourceInfo from the incoming Setup message to the outgoing Setup message. Otherwise, Net-Net SBC uses its own endpointType for the sourceInfo field.
noAliasinRCF	Net-Net SBC does not include any terminal alias in the RCF.
forceH245	Net-Net SBC initiates an H.245 connection after the call is connected. Otherwise, Net-Net SBC listens for an H.245 connection to be initiated by a remote endpoint.
useCPNInRAS	Net-Net SBC uses the calling party number (CPN) IE of the incoming call as the srcInfo of a RAS message sent in the outgoing call (such as an ARQ).
maintenanceProxy	Net-Net SBC registers interfaces on the enterprise side with a gatekeeper on the carrier side, and registers endpoints through the Net-Net SBC with a unique rasAddress. Interfaces on the enterprise side are associated with the carrier interfaces; you set this option on the carrier side.
convertPNTToE164	Net-Net SBC converts the address type partyNumber to dialedDigits (E.164). Conversion applies to sourceAddress, destinationAddress, and destExtraCallInfo aliases in Setup messages.

Option	Description
useCalledPNAsDestInfo	Net-Net SBC uses the H.225 called party number IE as the destinationInfo in ARQ and LRQ requests. Since translation rules can be applied to the Called Party Number, the option enables digit normalization for RAS requests. When not used, Net-Net SBC derives the destinationInfo field in RAS requests from the DestinationAddress field of the incoming Setup.
waitForIncomingH245	On the incoming leg, the Net-Net SBC does not send out its h245Address, but waits for the calling endpoint to send its H245Address. Applies to the outgoing call leg as well: The Net-Net SBC does not send out a Facility with startH245 reason and waits for the called endpoint to send its H245Address.
uniqueRRQSrcPort	Enables H.323 RAS Port Mapping. The Net-Net SBC uses the RAS port that it assigned in the rasAddress parameters of an RRQ message as the UDP source port of the outgoing RRQ. Because this feature is linked to the unique RRQ functionality, be aware of the following before you enable the feature: - Enabling H.323 RAS Port Mapping automatically enables the Net-Net SBC's unique RRQ functionality, eliminating the need for you to configure the latter as a separate option. - Enabling the unique RRQ functionality (by setting the uniqueRRQRASAddress option) does not automatically enable H.323 RAS Port Mapping.

H.323 Stack Monitoring

Previously, the Net-Net SBC provided SNMP monitoring of H.323 session agents but not of the H.323 stacks themselves. The H.323 stack/interface configuration now provides a way for you to set alarm thresholds on a per-stack basis. When enabled, this alarm system ties into the **Max. concurrent calls** value to send critical, major, or minor alarms when the number of calls approaches the threshold.

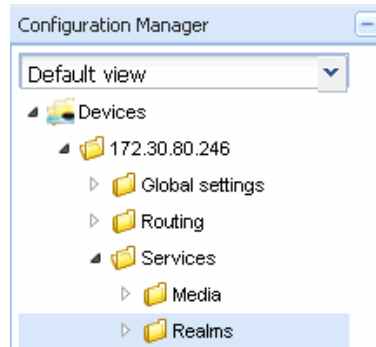
Each H.323 stack now has a threshold crossing alert (TCA) where you can set up three severity levels: critical, major, and minor. You can define one severity level or all three for each stack. To prevent the alarm from firing continuously as call volume through the stack varies, each severity level has a reset value below the TCA you set. In addition, each threshold value resets when:

- An alarm with a higher severity is triggered, or
- The built-in reset value for the threshold level is 1% less than the parameter value

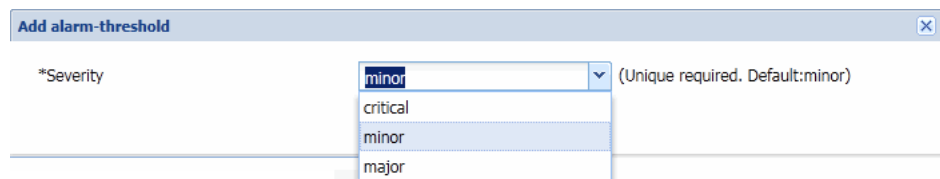
Configuring H.323 Stack Monitoring

To configure H.323 stack monitoring:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Click an H.323 service/protocol in the table and click **Edit**.
6. Scroll to the Alarm threshold table.
7. Click **Add**. The Add H.323 alarm threshold dialog box appears.
8. **Severity**—Click the type of severity level in the combination box for the alarm you want to define. This value is required, and defaults to **minor**.



9. Click **Apply**.
10. Select the alarm threshold from the table and click **Edit**.
11. **Value (%)**—Enter the percentage of the number of calls defined in the **Max. concurrent calls** parameter that triggers the alarm. For example, if you want to set a minor alarm to fire when the call rate through the stack reaches half the **Max. concurrent calls** value, enter **50** (meaning 50%). The default value for this parameter is 0, which disables the alarm.

Remember that if the number of calls falls to below 1% of the Max. concurrent calls threshold you set, the clear trap fires.

*Severity	minor
Value (%)	50

12. Click **Apply**.

H.323 Automatic Features

This section describes H.323 features that are automatically enabled on your Net-Net system. You do not have to configure special parameters to turn them on. Even though you do not have to turn these features on, this section describes what they do and how they work.

Alias Mapping

Alias mapping permits destination addresses to be modified by a gatekeeper.

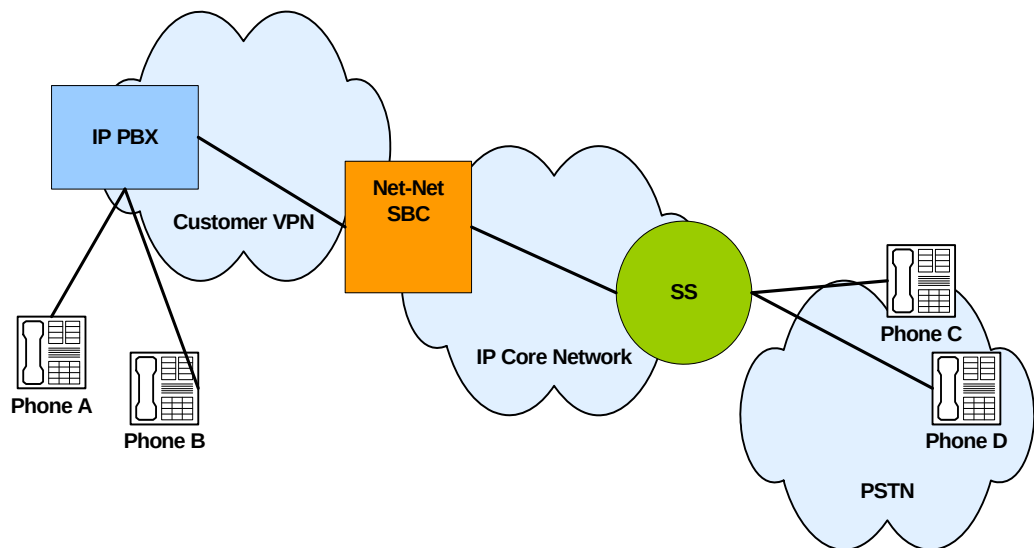
When sending an ARQ or an LRQ message to a gatekeeper, the Net-Net SBC sets the `canMapAlias` field in that message to `true`. This setting indicates that the Net-Net SBC accepts modified destination information from the gatekeeper. If the resulting ACF or LCF contains `destinationInfo` and/or `destExtraCallInfo` fields, then the Net-Net SBC copies that information respectively to the `destinationAddress` and `destExtraCallInfo` fields of the Setup message. In addition, if the `destinationInfo` is either type `e164` or type `partyNumber`, the Net-Net SBC copies the information into the `calledPartyNumber` information element (IE) of the Setup message, replacing the existing `calledPartyNumber` IE.

You do not need to configure special parameters for this feature; it is enabled automatically.

Call Hold and Transfer

The Net-Net SBC's H.323 call hold and transfer feature supports consultation in addition to call holder and transfer. This feature uses signaling procedures based on the ITU-T recommendations/H.323 specification for what it calls *third party initiated pause and rerouting*.

The following diagram shows how the Net-Net SBC is positioned to provide call hold and transfer support for H.323.



Call Hold and Transfer: Basic Call

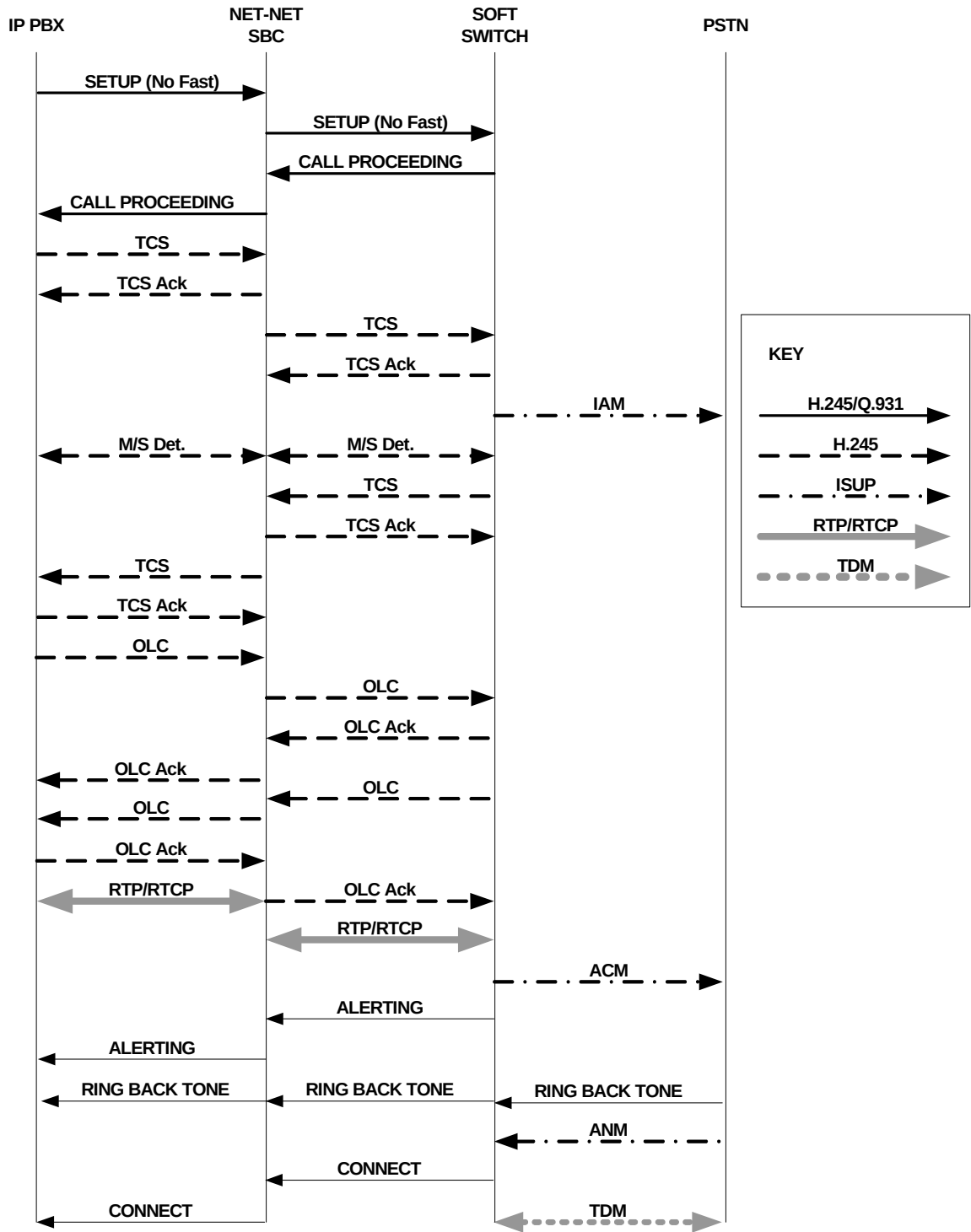
The following diagram show the signaling and media flows between the IP PBX and a softswitch. Note how the Net-Net SBC is position to mediate flows between the two devices.

In the Call Proceeding messages forwarded to the IP PBX, the Net-Net SBC uses a non-zero value to ensure that the IP PBX initiates an H.245 session. A progress indicator does not need to be included if the H.245 address is present in any of the following message types: Alerting, Progress, or Connect.

After the Net-Net SBC receives a Call Proceeding message from the softswitch that contains the H.245 address, the Net-Net SBC sends another Call Proceeding with its own H.245 address.

In the following call flow, the softswitch generates message to the gateway. These messages are:

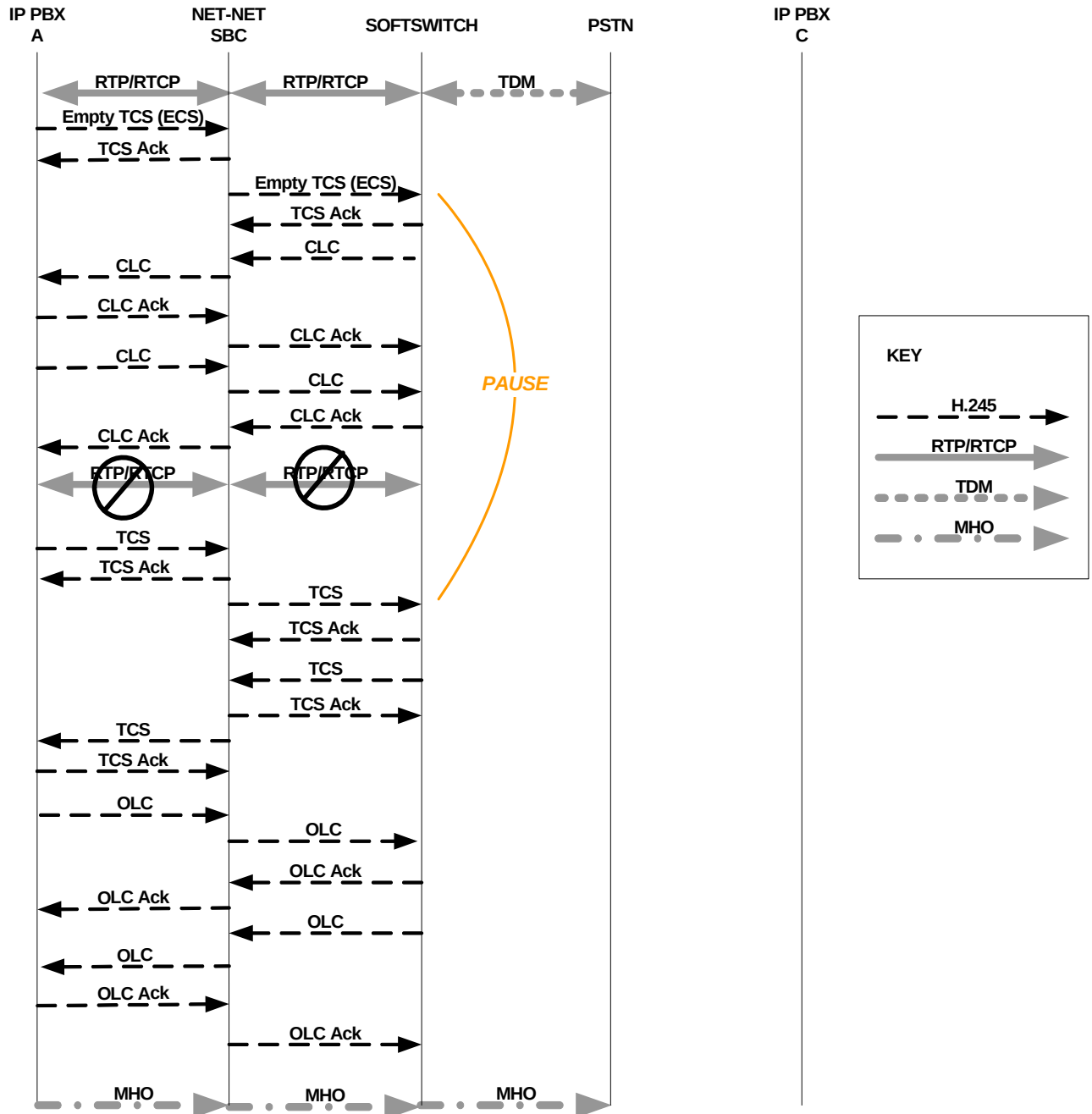
- Initial Address Message (IAM)
- Address Complete Message (ACM)
- Answer Message (ANM)



**Call Hold and Transfer:
Music on Hold**

The following diagram begins with the condition that IP PBX A is already connected with a gateway, with the Net-Net SBC and the softswitch positioned between the two.

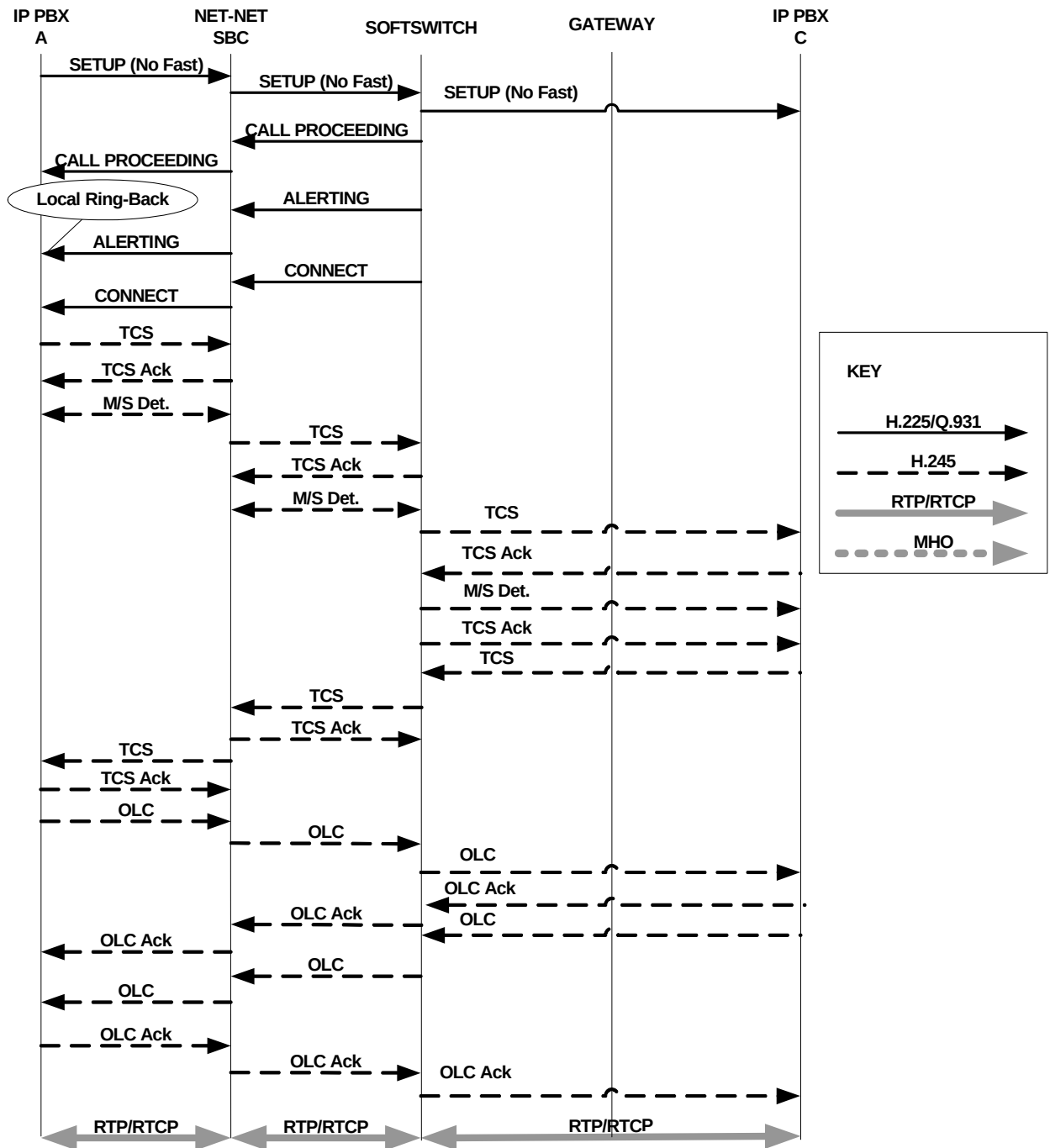
You can see in the call flow where the channels for transporting media are closed, and where the RTP/RTCP is stopped. This creates a pause for the call. With the Net-Net SBC mediating the process, IP PBX A and the softswitch exchange TCS and OLC messages that allow music on hold (MHO) to flow between IP PBX A and the gateway.

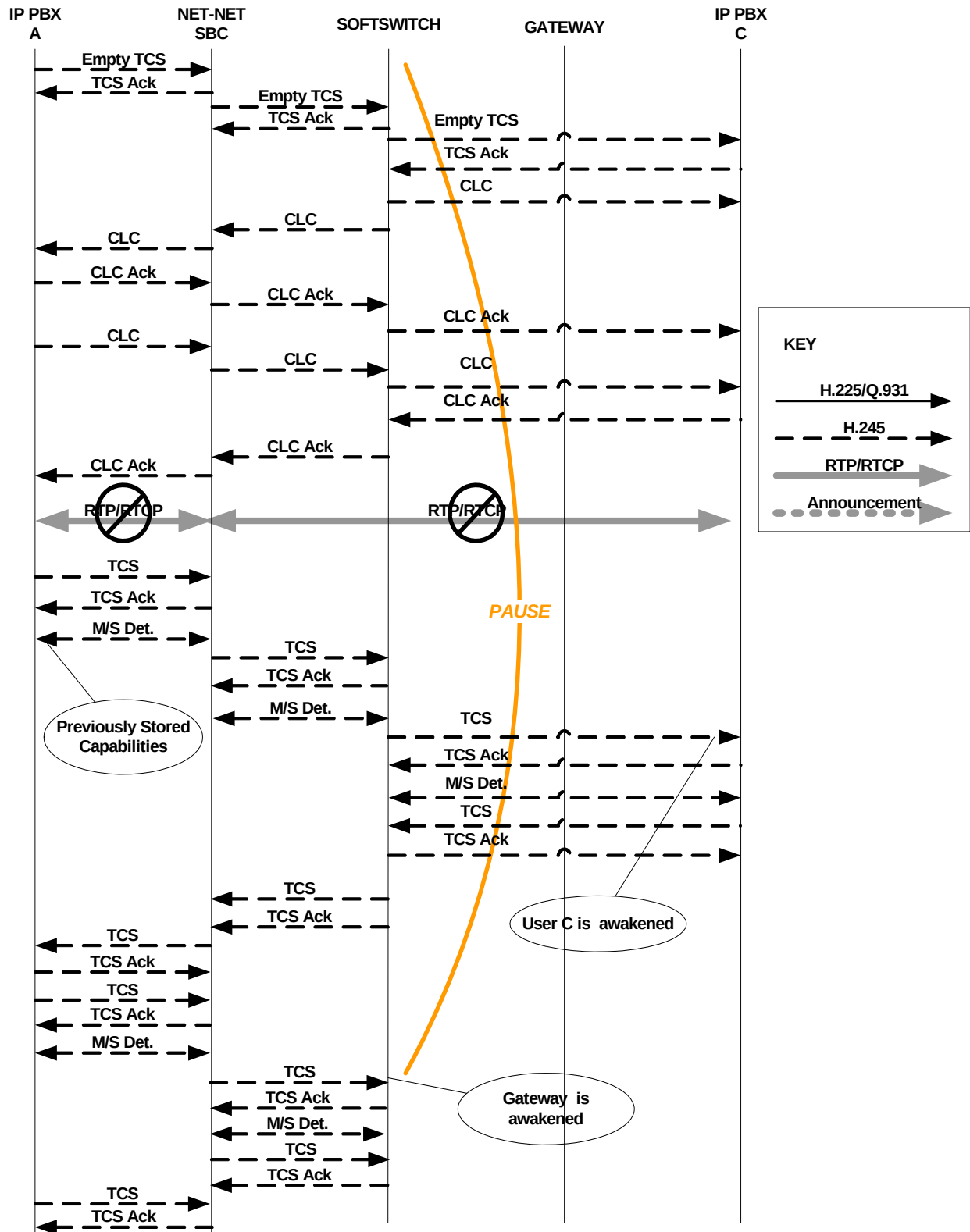


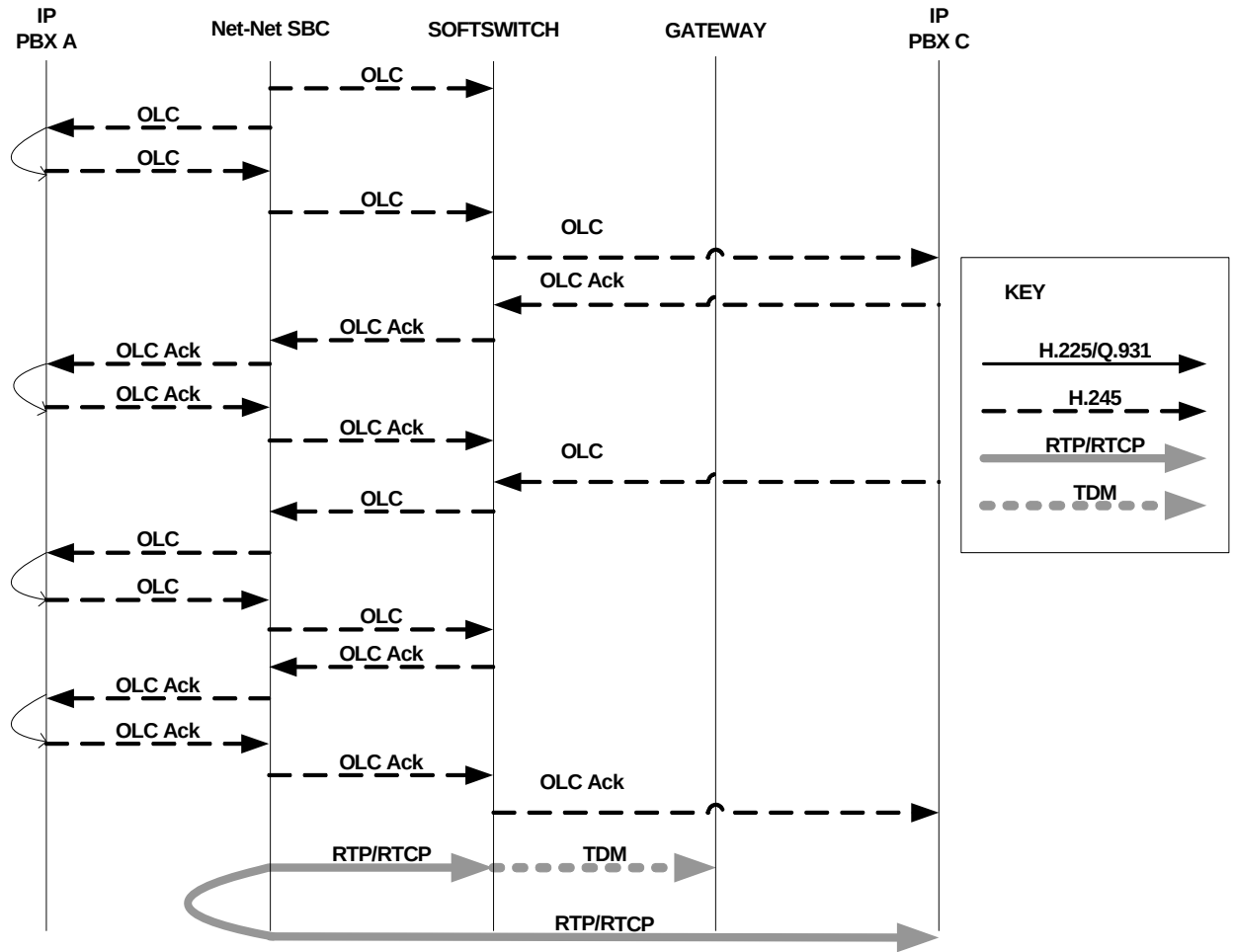
Call Hold and Transfer: Transfer

The following diagram shows how call transfer works on the Net-Net SBC for H.323. In this diagram, you can see:

- Where local ringback occurs
- Where the pause begins and ends
- Where users and gateways are awakened
- Where logical channels are opened and closed







Media Release for SS-FS Calls

When the Net-Net SBC routes a slow-start to fast-start call, it is possible for the same fast-start call to be routed back through the Net-Net SBC making for a hairpin flow. If it does become a hairpin flow, then the Net-Net SBC routes it to its destination as a fast-start to fast-start call. This can result in one-way media if:

- The destination of the hairpin call is in the same realm as the originating slow-start to fast-start call
- The realm reference in the first bullet item is configured to disable in-realm media management
- The called endpoint accepts the proposed fast-start logical channels

The enhancements to the Net-Net SBC's behavior described in this section show how the Net-Net SBC follows additional procedures when setting up a hairpin flow to avoid one-way media when media release occurs.

How It Works

For H.323 calls, the Net-Net SBC establishes media using the H.245 procedures described in the H.245 ITU-T recommendation: control protocol for multimedia communication. It also uses the Fast Connect procedure defined in the H.323 ITU-T recommendation: packet-based multimedia communication systems.

The latter ITU-T recommendation allows a calling endpoint to send a Setup message that contains a fastStart element, a sequence of OLC structures that describe the calling endpoint's proposed forward/reverse logical channels. If the called endpoint accepts this proposal, then logical channels are established.

When the Net-Net SBC translates a call originating in slow-start to fast-start, it uses a Fast Connect procedure in the outgoing leg by sending an outgoing Setup that includes a fastStart element with one or more OLC structures. But when the Net-Net SBC constructs this message, it is unaware of whether the call will become hairpinned or if media release will occur. Because it does not yet have this information, the Net-Net SBC sets the Network Address and the TSAP identifier in the OLC structures to the ingress IP address and port of a corresponding media flow allocated for media traveling between the calling and called endpoints. So if the called endpoint accepts the fastStart the Net-Net SBC proposes, the called endpoint would send its media to the Net-Net SBC. After acceptance, the Net-Net starts H.245 procedures on the slow-start side of the call to set up logical channels on that side. Then the Net-Net SBC updates the IP address and port of the media flows using OLC and OLCAck messages received from the calling endpoint.

This procedure works well for endpoints that are not in the same realm, or that are in the same realm for which media management is disabled, because each endpoint must send its media through the Net-Net SBC. When the endpoints are in the same realm and when media management is enabled, however, the Net-Net SBC must perform additional steps for media release in slow-start to fast-start calls.

To support media release in slow-start to fast-start calls, the Net-Net SBC performs a hold-and-resume procedure on the fast-start side. After it establishes channels on the slow-start side and if it detects media release being enabled, the Net-Net SBC sends an empty TCS to the fast-start side to put that side on hold. Then the called endpoint closes all the logical channels it previously opened in the Fast Connect procedure and stops transmitting to them. And the Net-Net SBC also closes its logical channels. Once the channels are closed, the Net-Net SBC resumes the call by sending a new, restricted TCS to the fast-start side. The restricted TCS only contains the receive and transmit capabilities of the codec types that the called endpoint accepted in the Fast Connect procedure, and it forces the called endpoint to re-open logical channels of the same codec types accepted in the Fast Connect procedure. Once it receives an OLC from the called endpoint, the Net-Net SBC sends an OLCAck with the Network Address and TSAP identifier for the logical channel from the calling endpoint. Then the Net-Net SBC re-opens logical channels (of the same codec types that it opened in the Fast Connect procedure). If the called endpoint has not changed its Network Address and TSAP identifier for its logical channels, media is re-established after the Net-Net SBC and the called endpoint exit the hold state. The last step is for the Net-Net SBC to re-send the full TCS message from the calling to the called endpoint to inform the called endpoint of the full capabilities of the calling endpoint.

Dependencies

This feature depends on the following assumptions:

- The H.323 endpoint supports the third-party-initiated pause and re-routing feature.
- The H.323 endpoint does not change its Network Address and TSAP identifier when it re-opens the logical channels.
- The H.323 endpoint does not immediately tear down the call when there is not established logical channel in the call.

Hold-and-Resume Procedure

The hold-and-resume procedure has three states:

- **Media Hold**—Starts when the Net-Net SBC sends the empty TCS to the called endpoint to put it on hold.

When it detects media release, the Net-Net SBC puts the called endpoint on hold. It can only do so if it has exchanged the TCS/TCSAck messages and completed master-slave determination with the calling endpoint.

When the Net-Net SBC receives a TCSAck in response to the empty TCS that it sent to the called endpoint, it closes the logical channels it opened as part of the Fast Connect procedure; the called endpoint likewise closes its logical channels. The two then exchange CLC and CLCAck messages, which signals the start of the Media Resume state.

- **Media Resume**—Starts when the Net-Net SBC sends a restricted TCS to resume the call.

The restricted TCS the Net-Net SBC sends contains only the receive/transmit capabilities of the codec types previously accepted by the called endpoint in the Fast Connect procedure. This forces the called endpoint to re-open logical channels of the same codec type that were previously accepted in the Fast Connect procedure.

After sending this TCS, the Net-Net is ready (as specified in the ITU-T recommendations) to take part on the master-slave determination (MSD) process. However, the called party and not the Net-Net SBC initiates the MSD if it is required. The MSD is completed if necessary. Alternately, the called endpoint can start to re-open its logical channels. When it receives the first OLC from the called endpoint, the Net-Net SBC also starts to re-open its logical channels.

- **Media Complete**—Starts when all the logical channels that the Net-Net SBC re-opens are acknowledged by the called endpoint.

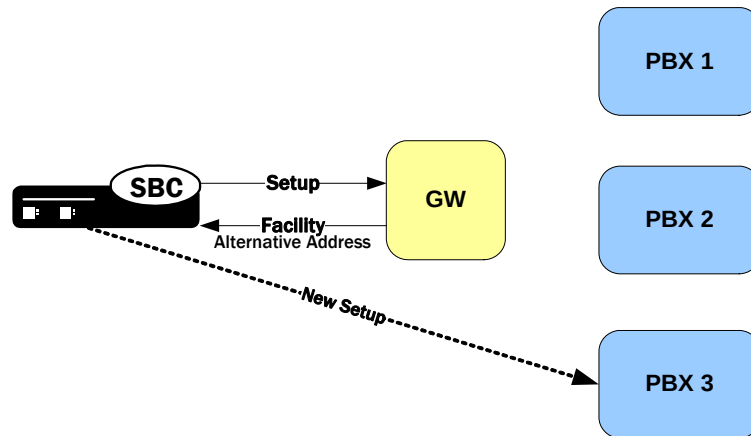
When it enters the Media Complete state, the Net-Net SBC updates the called endpoint with the full capabilities of the calling endpoint by sending the full TCS.

H.323 and IWF Call Forwarding

Net-Net SBC's H.323 and IWF call forwarding is supported for H.323 calls and for calls initiated in SIP that require interworking to H.323.

In the following diagram you can see that the Net-Net SBC sends the initial Setup message to the gateway, and the gateway returns the Facility message with an alternate address for forwarding. Rather than engaging in its former behavior, the Net-Net SBC now releases the call with the gateway and sends a new Setup to the alternate address from the Facility message.

This new Setup up has no effect on the first call leg, which remains connected.



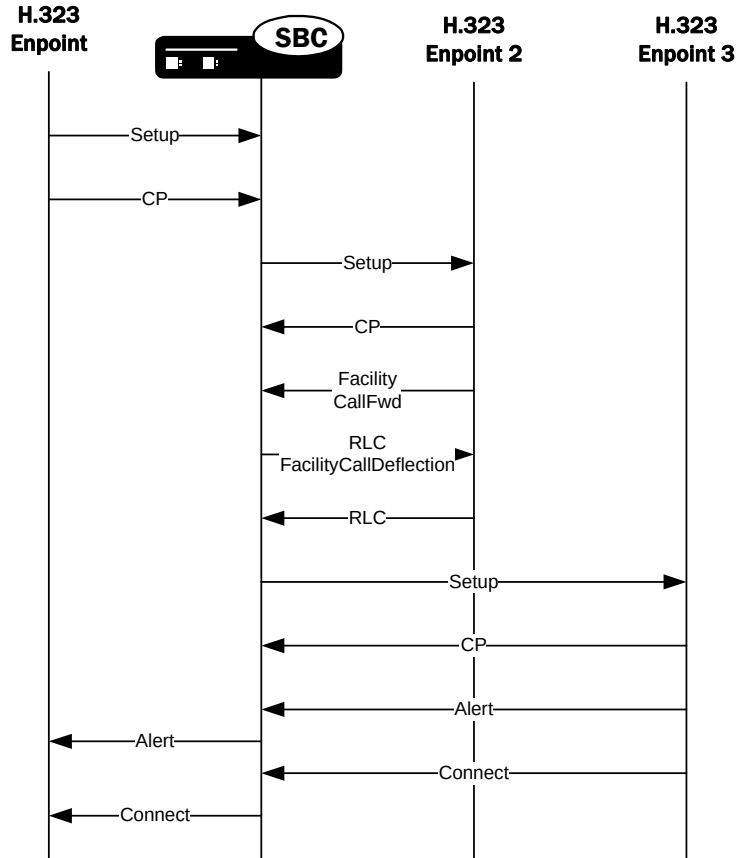
How It Works

When it receives a Facility message with the reason CallForwarded, the Net-Net SBC looks for an alternate transport address in the Facility's alternativeAddress or alternativeAliasAddress element. The Net-Net SBC releases the egress call with the reason facilityCallDeflection. Then it takes one of two courses of action:

- If it does not find an alternative address, the Net-Net SBC releases the ingress call (with the reason facilityCallDeflection).
- If it finds an alternative address and the egress call has not been alerted or answered, the Net-Net SBC at this point tries to initiate a new egress call. The Net-Net SBC uses the alternative alias address to populate the calledPartyNumber information element (IE) and the destination address of the new Setup.

H.323 Sample Call Flow

The following diagram shows how the H.323 Call Forwarding feature works in a purely H.323 environment.



H.323 Accounting

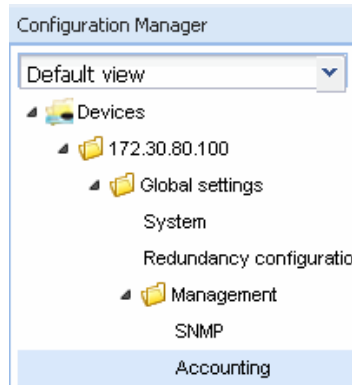
This section tells you how to access and set the values to configure accounting for H.323. It also provides sample configurations for your reference. To use the Net-Net system in conjunction with external RADIUS (accounting) servers to generate CDRs and provide billing services requires, you need to configure account configuration and account server list.

Configuring H.323 Accounting

This section describes how to configure H.323 accounting on your Net-Net SBC from the Net-Net Central.

To configure H.323 accounting:

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. **State**—Set the state to **enabled** from the combination box.

State enabled ▼

5. **Host name**—Enter **localhost** as the hostname of this Net-Net system to ensure the accounting configuration works properly. Entries are in FQDN format.
6. **Port**—Enter the UDP port number from which RADIUS messages are sent.

Host name localhost

Port 1813

Messages

1. Scroll to Messages.
2. **Start**—Set when a RADIUS Start record should be generated from the combination box. Choices are:
 - **Invite**—Upon an H.323 Setup message
 - **None**—RADIUS start message should not be generated
 - **OK**—Upon an H.323 Connect message

Start OK ▼

OK
 None

3. For the Interim parameter, click **Add**. The Add Interim dialog box appears.

4. **Event**—Select OK from the combination box so that RADIUS Interim record generation corresponds to an H.323 Connect message.

5. Click **Apply**. The interim appears in the Interim list.

Interim

6. **Transmission timing**—Select enabled from the combination box to transmit account information at the start of a session (Start), during the session (Interim), and at the close of a session (Stop). The transmitted accounting information of a single session might span a period of hours and be spread out among different storage files.

Transmission timing

Intermediate Period

You can set how often the Net-Net SBC generates periodic interim records for H.323. The periodic timer (set to the value you specify in the accounting configuration) is dynamically created when the Net-Net SBC receives a 200 OK response to an INVITE message. The Net-Net SBC deletes the timer when the session is terminated.

See the *Net-Net RADIUS Guide* for more details.

To set the timer for periodic interim records:

1. Scroll to Timer and message queue limits.
2. **Period between interim CDRs (sec)**—Enter the amount of time in seconds between generating periodic interim records during a SIP or H.323 call. This parameter defaults to zero, which is not a valid value.

Period between interim CDRs(sec)

Timers and Message Queue Limits

1. Scroll to Timers and message queue limits.
2. **Max. time - message retransmit (sec)**—Enter time in seconds the Net-Net SBC should try to send each RADIUS message or retain the default value of 60. During this delay, the Net-Net system can hold a generic queue of 4096 messages. Range of values is 0 to $2^{32}-1$.
3. **Max. message queue before failover (#)**—Enter the number of messages that must accumulate in the queue before a failover situation is enacted or retain the default value of 100. Range of values is 1 to 4096.

Max time-message retransmit(sec)	60
Max. message queue before failover (#)	100

Once this value is exceeded, the Net-Net system attempts to send its accounting messages, including its pending messages, to the next accounting server on its configured list.

Local CDR Generation and Storage

1. Click the Push and file based accounting tab at the top of the content area.
2. **Output to file**—Select enabled from the combination box to enable the output of comma-delimited CDRs (generated from RADIUS records) for this Net-Net SBC. To use FTP push, this parameter must be enabled.
3. **File path**—Enter the path to use on the Net-Net SBC for file storage from these two options:
 - /ramdrv
 - /ramdrv/logs

You must configure this path or the CDR push feature will not work. To use FTP push, you must configure a usable path.
4. **Maximum file size**—Enter the maximum file size in bytes permitted for each CDR file. The default and minimum value is 1000000. The maximum value is 10^8 .
5. **Maximum files**—Enter the maximum number of files to be stored on the Net-Net SBC at one time. You can configure the Net-Net SBC to store as few as one file or as many as 10. The default is 5.
6. **File rotaional interval**—Set the file rotation interval how often in minutes you want to rotate the stored files; the Net-Net SBC will overwrite the oldest file

first. The minimum rotation time is 2 minutes. The rotation time is only used when CDR push is enabled.

Local CDR file generation/storage

Output to file	enabled
File path	
Maximum file size(bytes)	1000000
Maximum files	5
File rotation interval(minutes)	0
File compression	disabled
File delete alarm	disabled

7. **Push CDR file via FTP**—Select enabled from the combination box to enable the FTP push feature.
8. **FTP IP address**—Enter the IP address of the FTP server to which the Net-Net SBC will send files.
9. **FTP port**—Enter the TCP port on the FTP server to which the Net-Net SBC will send files; the minimum is 1025 and the maximum is 65535.
10. **FTP user**—Enter the username for the FTP server to which the Net-Net SBC will send files.
11. **FTP password**—Enter the password corresponding to the FTP username.
12. **File remote path**—Enter the file path on the FTP server to which the Net-Net will send files.
13. **Multiple push receiver usage server**—Set the strategy for the Net-Net SBC to use when selecting from multiple push receivers
 - **hunt**—The Net-Net SBC selects the push receiver from the available list according to the priority level
 - **failover**—The Net-Net SBC selects the push receiver based on priority level and continues to use that same push receiver until it fails over
 - **roundrobin**—The Net-Net SBC selects push receivers systematically one after another, balancing the load among all responsive push receivers
 - **fastesttrtt**—The Net-Net SBC selects the push receiver based on best average throughput. For this situation, throughput is the number of bytes transferred divided by the response time. The system uses a running average of the five most recent throughput values to accommodate for network load fluctuations
14. **Maximum wait time for timeout for push receivers**—Enter the amount of time in seconds to wait before the Net-Net SBC declares a push receiver to have failed over.

15. **CDR output redundancy**—Select enabled from the combination box to enable redundant storage of comma-delimited CDR files.

FTP parameters

Push CDR file via FTP	enabled ▼
FTP IP address	2.0.0.0
FTP port	21
FTP user	acmeuser
FTP password	••••••••
File remote path	/sdRADIUS
Multiple push receiver usage strategy	Hunt ▼
Maximum wait time for timeout for push receivers (sec)	120
CDR output redundancy	enabled ▼

Accounting Servers

1. Click the Radius accounting tab at the top of the content area.
2. Click **Add** to add a new RADIUS server.

RADIUS accounting servers

Hostname ▼	Port	State
qwed	1813	enabled
Add Edit Delete		

The Add Accounting server dialog box appears.

3. **Host name or IP address**—Enter the hostname or IP address of the RADIUS server you are configuring.
4. **IP port**—Enter the IP port number used to communicate with this RADIUS server.

5. Click **Apply**.
6. Select the account server from the table and click **Edit**.
7. **State**—Select the operational state of this RADIUS server from the combination box.
8. **VSA bundling**—Select enabled from the combination box to enable the account server to bundle the VSAs within RADIUS accounting messages. Bundling means including multiple VSAs within the vendor value portion of the message.
9. **Min. round trip time threshold (ms)**—Enter the minimum roundtrip of an accounting message in milliseconds. Minimum is 1025 and maximum is 65535.

A round trip consists of the following:

- The Net-Net system sends an accounting message to the account server.
- The account server processes this message and responds back to the Net-Net system.

If the fastest RTT is the strategy for the account configuration, the value you enter here can be used to determine the order of preference (if all the configured account servers are responding in less than their minimum RTT).

10. **Max. time ACK (sec)**—Enter the maximum time in seconds the Net-Net system waits when accounting messages are pending without a response before this account server is set as inactive for its failover scheme. Range is 1 to 300.
11. **Min. time-reinstate delay (sec)**—Enter the time in seconds the Net-Net system waits after declaring an accounting server inactive before resending an accounting message to that same accounting server. Range is 1 to 300.

Once this timer value is exceeded, the Net-Net system marks the unresponsive account server as disabled in its failover scheme. When a server connection is marked as inactive, the Net-Net system attempts to restart the connection and transfers pending messages to another queue for transmission. RADIUS messages might be moved among different account servers as servers become inactive or disabled.

12. **Priority**—Enter the number corresponding to the priority for this account server to have in relation to the other account servers to which you send traffic. The default is 0, meaning there is no set priority.
13. **Domain name suffix**—Enter the domain name suffix for payload strings constructed as a domain name. If your entry does not include a dot, the system appends one.

14. **Origin realm**—Select the origin realm from the combination box.

State	enabled ▼
VSA bundling	enabled ▼
*Host name or IP address	packet.com
*IP port	1813
Min. round trip time threshold (ms)	250
Max.time ACK (sec)	60
Min.time-reinstate delay (sec)	30
Priority	0
Domain name suffix	
Origin realm	▼

15. Click **Apply** to save your information and close the Accounting server dialog box.

The Accounting server dialog box closes and the new accounting server entry appears in the Names & state list.

Hostname	Port	State
qwed	1813	enabled
packet.com	1813	enabled

16. **Multiple server usage strategy**—Select the strategy you want used to select the accounting servers to which the Net-Net system will send its accounting messages from the combination box.

The following table lists the available strategies:

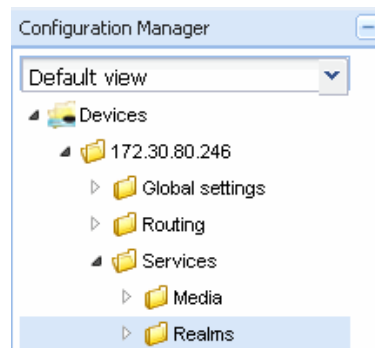
Strategy	Description
hunt	Selects accounting servers in the order in which they are listed. If the first accounting server is online, working, and has not exceeded any of the defined constraints, all traffic is sent to it. Otherwise the second accounting server is selected. If the first and second accounting servers are offline or exceed any defined constraints, the third accounting server is selected. And so on through the entire list of configured servers.
failover	Uses the first server in the list of predefined accounting servers until a failure is received from that server. Once a failure is received, it moves to the second accounting server in the list until a failure is received. And so on through the entire list of configured servers.
round robin	Selects each accounting server in order, distributing the selection of each accounting server evenly over time.
fastest round trip time	Selects the accounting server that has the fastest round trip time (RTT) observed during transactions with the servers (sending a record and receiving an ACK).
fewest pending	Selects the accounting server that has the fewest number of unacknowledged accounting messages (that are in transit to the Net-Net system).

- Click **Apply** to complete configuring H.323 accounting.

Generating CDRs for a Realm

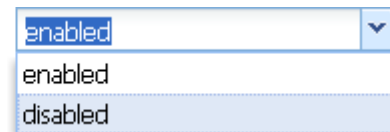
To generate CDRs for a realm:

- Expand the Configuration Manager slider > Services.



- Click Realms. The Realm table appears in the content area.
- Select a SIP realm in the table to select it and click **Edit**.
- Generate CDRs for this realm**—Select enabled from the combination box to generate CDRs for the realm.

Generate CDRs for this realm



- Click **Apply**.

Additional Information

This section contains detailed tables to use as a reference when you are learning about H.323 features or when you are configuring them.

About Payload Types

You set the payload type when you are configuring a media profile to support Slow Start to Fast Start Translation.

When you configure media profiles, you might need set the payload type to identify the format in the SDP *m* lines. For RTP/AVP, the default transport method of a media profile configuration, this will be the RTP payload type number. Newer codecs have dynamic payload types, which means that they do not have an assigned payload type number.

When you use RTP/AVP as the transport method, you should only set the payload type when there is a standard payload type number for the encoding name; otherwise, leave the payload type blank.

The Net-Net SBC uses the payload type value to determine the encoding type when SDP identifies the standard payload type in the *m* line, but does not include an *a=rtpmap* entry. These are two equivalent SDPs:

1. *c=IN IP4 192.0.2.4*
m=audio 0 RTP/AVP 0
2. *c=IN IP4 192.0.2.4*
m=audio 0 RTP/AVP 0
a=rtpmap:0 PCMU/8000

The first does not include the RTP map entry, but uses the standard payload type of 0. If the Net-Net SBC receives an SDP like the first, it uses the payload type 0 to locate the corresponding media profiles configuration. When an *a=rtpmap* is present, the Net-Net SBC uses the encoding name in the *a=rtpmap* line to find the media profile configuration and does not consider the payload type number.

Payload Types for Standard Audio and Visual Encodings

The following is a table of standard audio and visual payload encodings defined in H. Schulzrinne, GND Fokus, "RTP Profile for Audio and Visual Conferences with Minimal Control," RFC 1890, and in the *RTP Parameters* document in IANA's Directory of Generally Assigned Numbers.

Payload Type	Encoding Name	Audio (A)/Visual (V)	Clock Rate (Hz)
0	PCMU	A	8000
1	1016	A	8000
2	G721	A	8000
3	GSM	A	8000
4	G723	A	8000
5	DVI4	A	8000
6	DVI4	A	16000
7	LPC	A	8000
8	PCMA	A	8000

Payload Type	Encoding Name	Audio (A)/Visual (V)	Clock Rate (Hz)
9	G722	A	8000
10	L16	A	44100
11	L16	A	44100
12	QCELP	A	8000
13	reserved	A	
14	MPA	A	90000
15	G728	A	8000
16	DVI4	A	11025
17	DVI4	A	22050
18	G729	A	8000
19	reserved	A	
20	unassigned	A	
21	unassigned	A	
22	unassigned	A	
23	unassigned	A	
dyn	GSM-HR	A	8000
dyn	GSM-EFR	A	8000
dyn	L8	A	var.
dyn	RED	A	
dyn	VDVI	A	var.
24	unassigned	V	
25	CelB	V	90000
26	JPEG	V	90000
27	unassigned	V	
28	nv	V	90000
29	unassigned	V	
30	unassigned	V	
31	H261	V	90000
32	MPV	V	90000
33	MP2T	AV	90000
34	H263	V	90000
35-71	unassigned	?	
72-76	reserved for RTCP conflict avoidance	N/A	N/A
77-95	unassigned	?	

Payload Type	Encoding Name	Audio (A)/Visual (V)	Clock Rate (Hz)
96-127	dynamic	?	
dyn	BT656	V	90000
dyn	H263-1998	V	90000
dyn	MP1S	V	90000
dyn	MP2P	V	90000
dyn	BMPEG	V	90000

About RAS Message Treatment

When you enabled the H.323 Registration Proxy, the Net-Net SBC modifies and deletes certain fields as outlines in the table below. The Net-Net SBC sends on any fields that are not listed in this table without modifying or deleting them.

Note: Although the Net-Net SBC forwards a field, it does not always support the feature related to that field.

Field Name	Message	Deleted	Modified	Value Used in Modification
alternateEndpoints	RRQ, URQ, ACF	X		
alternateGatekeeper	RCF, URQ	X		
altGKInfo	RRJ, URJ, DRJ	X		
alternateTransportAddresses	RRQ, ARQ, ACF	X		
callModel	ARQ		X	direct
	ACF		X	gatekeeperRouted
callSignalAddress	RRQ		X	Mapped virtual CSA allocated by the Net-Net SBC for registering the endpoint.
	RCF, ARJ		X	CSA of gatekeeper stack
	URQ		X	If URQ is from an endpoint, endpoint's mapped virtual CSA. If URQ is from a gatekeeper, real CSA of endpoint.
destCallSignalAddress	ARQ, ACF	X		
destinationInfo.transportID	ARQ, ACF	X		
destExtraCallInfo.trasportID	ARQ, ACF	X		
discoveryComplete	RRQ		X	TRUE

Field Name	Message	Deleted	Modified	Value Used in Modification
endpointAlias.transportID	URQ	X		
endpointAliasPattern.Wildcard.transportID	URQ			
featureServerAlias.transportID	RCF	X		
gatekeeperIdentifier	RRQ		X	Gatekeeper identifier of the gateway stack, either configured in the H.323 gateway stack or discovered dynamically.
maintainConnection	RRQ, RCF		X	FALSE
multipleCall	RRQ, RCF		X	FALSE
preGrantedARQ.alternateTransportAddresses	RCF	X		
preGrantedARQ.useSpecifiedTransport	RCF	X		
rasAddress	RRQ		X	Mapped virtual RAS address allocated by the Net-Net SBC for registering endpoint
remoteExtensionAddress.transportID	ARQ, ACF	X		
srcCallSignalAddress	ARQ	X		
srcInfo.transportID	ARQ	X		
supportedH248Packages	RRQ	X		
supportsAltGK	RRQ	X		
supportedPrefixes.prefix.transportID	RCF, URQ	X		
terminalAlias.transportID	RRQ	X		
terminalAliasPattern.wildcard.transportID	RRQ	X		
willRespondToIRR	RCF, ACF	X		

Field Name	Message	Deleted	Modified	Value Used in Modification
willSupplyUIEs	RRQ, ARQ			
uuiesRequested	ACF			
setup			X	FALSE
callProceeding			X	FALSE
connect			X	FALSE
alerting			X	FALSE
information			X	FALSE
releaseComplete			X	FALSE
facility			X	FALSE
progress			X	FALSE
empty			X	FALSE
....				
status			X	FALSE
statusInquiry			X	FALSE
setupAcknowledge			X	FALSE
notify			X	FALSE

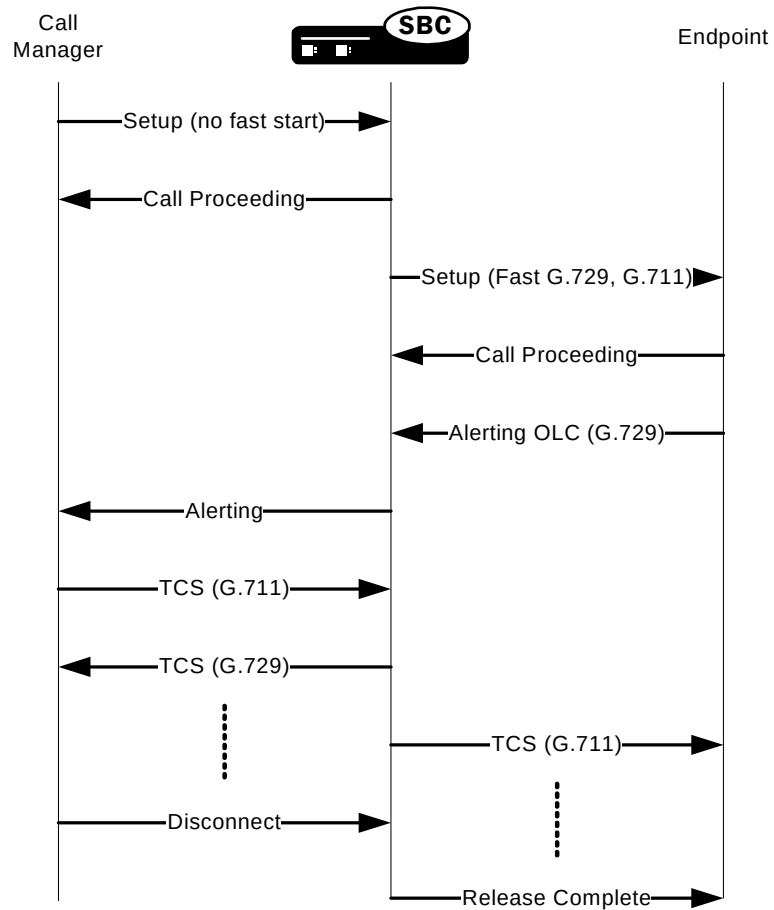
H.323 Codec Fallback

In the global H.323 configuration, you can enable a parameter that allows the Net-Net SBC to renegotiate—or fallback—to the preferred codec used in an incoming terminal capability set (TCS) from the slow-start side of a slow-start to fast-start H.323 call. When enabled, the Net-Net SBC performs this renegotiation when it detects a mismatch between the codec used in the open logical channel (OLC) opened on the fast-start side of the call, and the codec specified by the slow-start side.

Previous Behavior

In previous releases of the Net-Net OS, the Net-Net SBC opened a channel using the codec specified by the northbound side. Since the call manager had specified another preferred codec, the result was a codec mismatch leading to a dropped call.

The following diagram shows how codec mismatches ended in dropped calls.

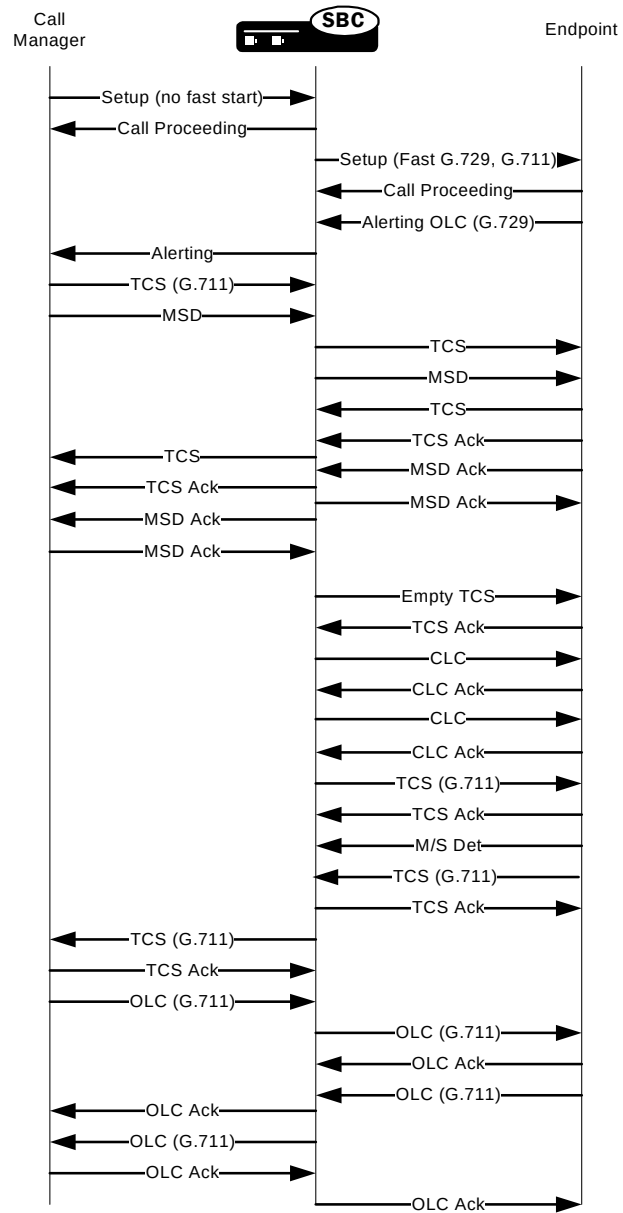


New Behavior

With H.323 codec fall back enabled, the Net-Net SBC attempts to use the preferred codec that the slow-start side of the call specifies. The Net-Net SBC determines matching based on the incoming TCS from the slow-start side and the OLC on the egress side. If the codecs do not match, the Net-Net SBC sends an empty TCS on the egress side and closes the logical channels on the outgoing side of the call.

To trigger a new capabilities exchange, the Net-Net SBC forwards the TCS from the ingress side of the call to the egress endpoint. Then the TCS from the egress endpoint is propagated to the ingress endpoint, and the logical channels are opened.

The following diagram shows a call scenario using the H.323 codec fallback feature.

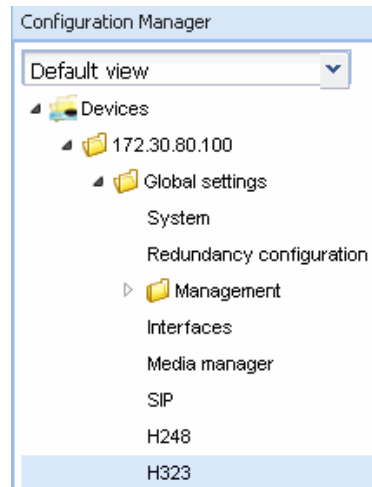


Net-Net Central Instructions and Examples

Note that you configure this feature for your global H.323 configuration, so it affects all H.323 traffic on your system.

To enable H.323 codec fallback:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. **Codec fallback**—Select enabled from the combination box.

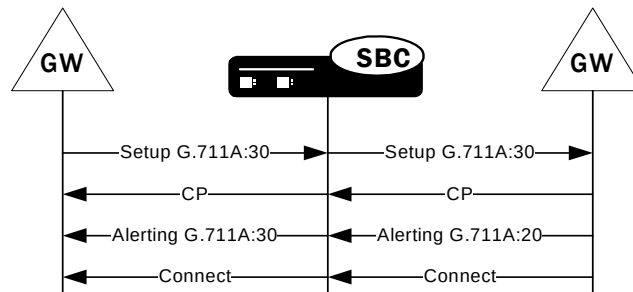


5. Click **Apply**.

H.323/TCS Media Sample Size Preservation

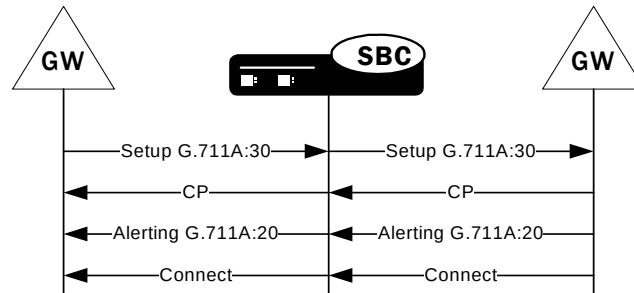
For H.323 fastStart calls, the Net-Net SBC can now be configured to preserve the packetization interval from the called gateway if it differs from the one offered in the Setup message the calling gateway sent.

In prior releases and in accordance with the ITU H.323 recommendation, the Net-Net SBC changes the packetization rate to the one used by the calling gateway if the one offered by the called gateway differs. In the following example, this means that the Net-Net SBC replaces the packetization interval of 20 with 30 before it forwards the Alerting message to the calling gateway.



However, not all H.323 elements comply with the ITU recommendation. Since some network elements do modify the packetization rate in the dataType element, this behavior is now configurable.

When you enable media sample size preservation, the Net-Net SBC allows the packetization rate to be modified and forwards on the modified dataType element to the calling gateway. In the following example, you can see that the Net-Net SBC forwards the called gateway's Alerting with the packetization interval of 20 despite the fact that the calling gateway's Setup specified 30.



Note that the calling endpoint might or might not work with the modified dataType.

You can enable this feature for the global H.323 configuration so that it applies to all H.323 fastStart calls, or you can enable it on a per-H.323 interface (stack) basis. When you enable this feature for an individual H.323 interface (stack), the Net-Net SBC performs media sample size preservation for calls egressing on that interface.

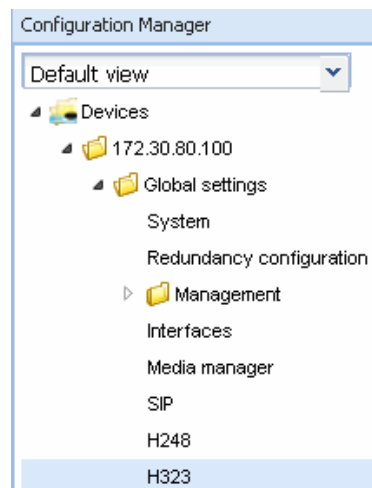
Configuring Media Sample Size Preservation

Enabling Media Sample Size Preservation: Global H.323 Configuration

This section shows you how to configure media sample size preservation for the global H.323 configuration and for an individual H.323 interface (stack).

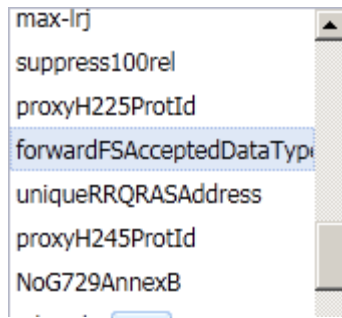
To enable media sample size preservation for the global H.323 configuration:

1. Expand the Configuration Manager slider > Global settings.

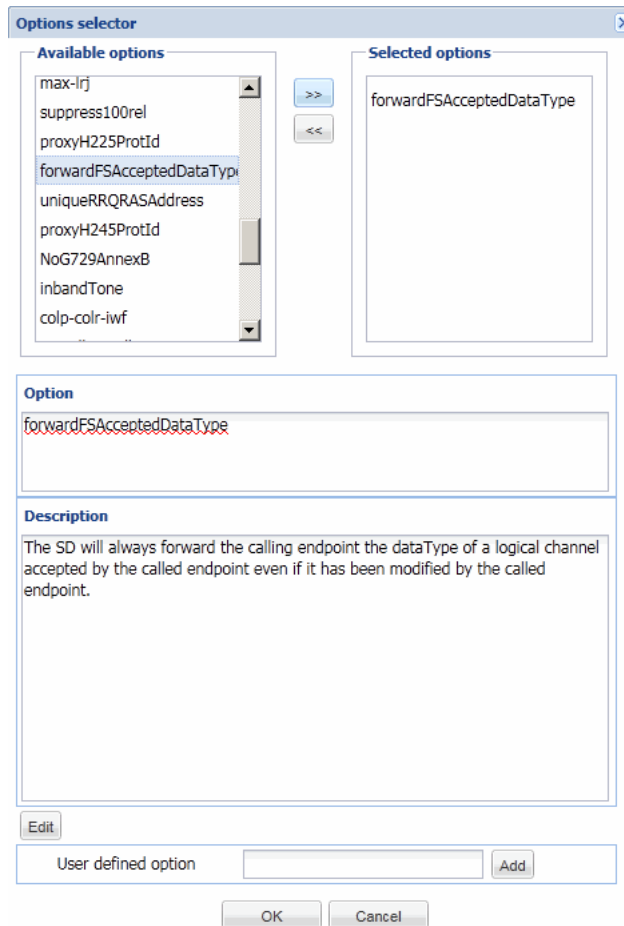


2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.

4. Click **Options**. The Options dialog box appears.
5. Click **forwardFSAcceptedDataType** in the Available options list to select it.



6. Click  to move the option to the Selected options list.

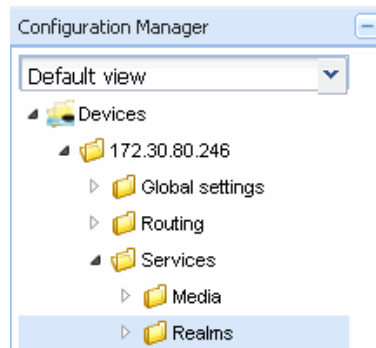


7. Click **OK**.
8. Click **Apply**.

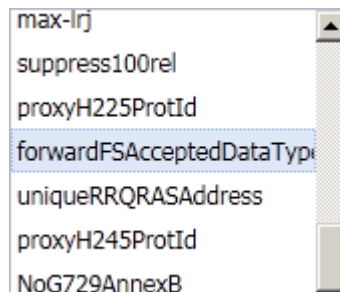
Enabling Media Sample Size Preservation: Individual H.323 Interface

To enable media sample size preservation for an individual H.323 interface:

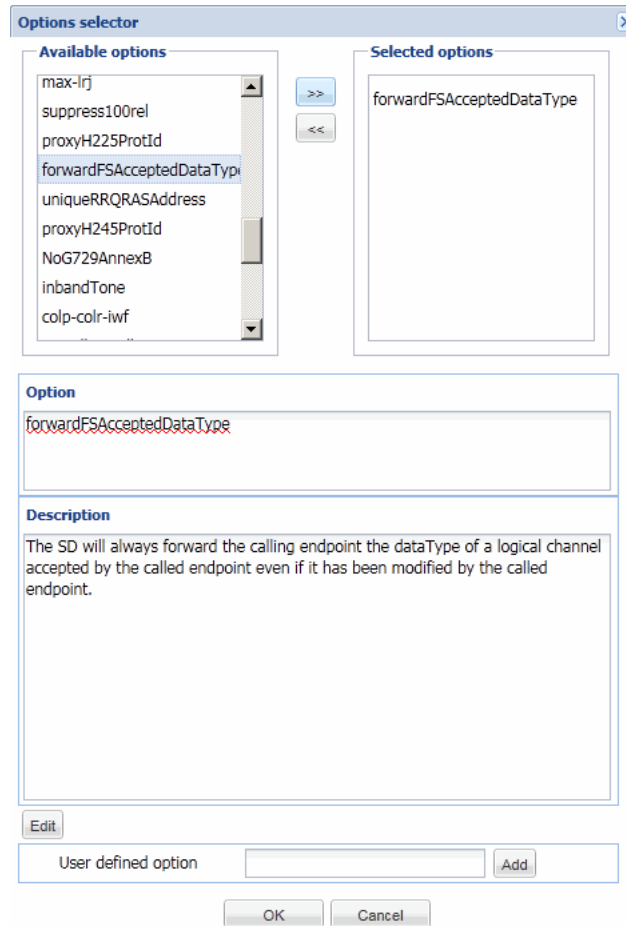
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 interface tab at the top of the content area.
7. Click **Options**. The Options dialog box appears.
8. Click **forwardFSAcceptedDataType** in the Available options list to select it.



9. Click  to move the option to the Selected options list.



The **Options selector** dialog box is shown. It has two main panes: **Available options** and **Selected options**. In the **Available options** list, `forwardFSAcceptedDataType` is selected. Below the panes, the **Option** field displays `forwardFSAcceptedDataType` and the **Description** field contains the text: "The SD will always forward the calling endpoint the dataType of a logical channel accepted by the called endpoint even if it has been modified by the called endpoint." At the bottom, there is an **Edit** button, a **User defined option** text box, an **Add** button, and **OK** and **Cancel** buttons.

10. Click **OK**.
11. Click **Apply**.

H.323-TCS: H.245 Support for H.264 and G722.1

The Net-Net SBC supports the H.264 video codec and the G722.1 audio codec. Especially useful for customer video product offerings in which the Net-Net SBC is deployed, this support further allows the Net-Net SBC to increase ease of use by supporting private addressing. Without this feature enabled (the Net-Net SBC's previous behavior), the Net-Net SBC required deployment for IANA registered IP addresses—despite the fact that IP VPNs allow for RFC 1918 private addressing.

Configuring

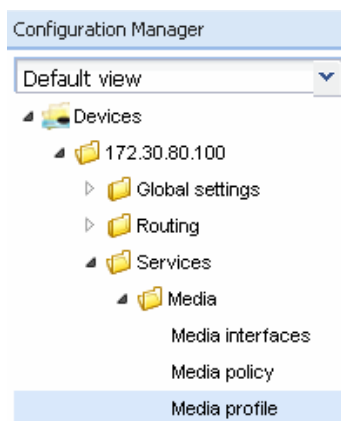
You need to set up media profile configurations appropriately. Media profiles allow you to set the configuration either as "generic video" or "generic audio."

H.245 provides for defining new capabilities that are described as H.245 generic capabilities (GenericCapability), which the Net-Net SBC now supports using the H.245 GenericCapability structure. H.264 and G.722.1 are the first codecs the Net-Net SBC offers that use this mechanism.

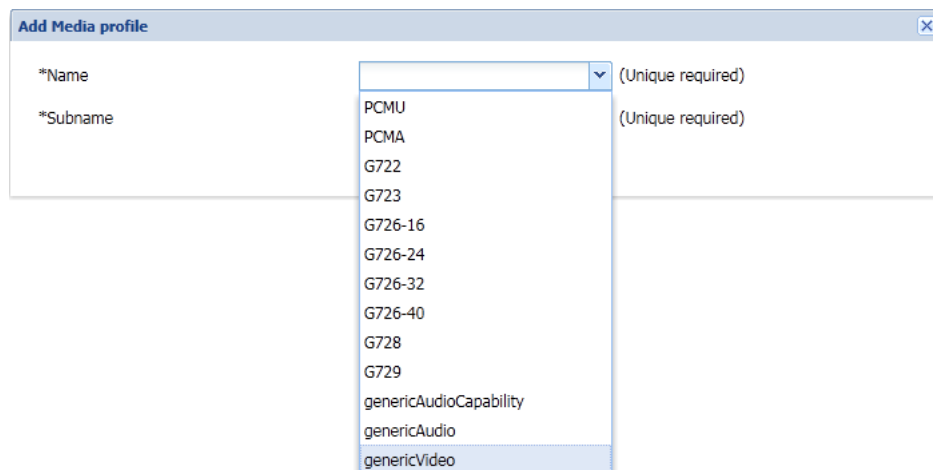
Creating a Generic Video Media Profile

To set a media profile for generic video support:

1. Expand the Configuration Manager slider > Services > Media.



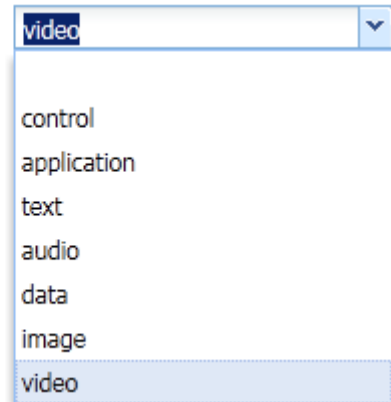
2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**. The Add Media profile dialog box appears.
4. **Media profile name**—Choose **genericVideo** from the combination box to set the name of the generic video media profile.



5. **Subname**—Enter a subname to distinguish this genericVideo media profile.
6. Click **Apply**. The genericVideo media profile appears in the Media profile table.
7. Select the genericVideo profile in the Media profile table and click **Edit**. The Edit Media profile dialog box appears.

8. **Media type**—Choose video from the combination box as the media type to use for this media profile.

Media type



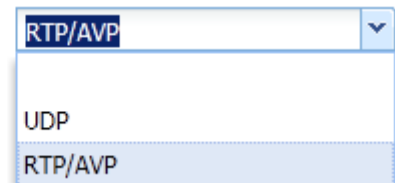
9. **Payload type**—Enter the payload type to use for the generic video media profile.

Payload type

11

10. **Transport protocol**—Choose the transport type to use for the generic video media profile from the combination box.

Transport protocol

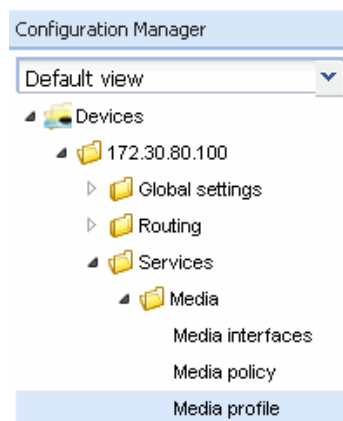


11. Complete the rest of the media profile configuration as needed.
12. Click **Apply**.

Creating a Generic Audio Media Profile

To set a media profile for generic audio support:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**. The Add Media profile dialog box appears.

4. **Media profile name**—Choose **genericAudio** from the combination box to set the name of the generic video media profile.

Add Media profile

*Name: genericAudio (Unique required)

*Subname: PCMU, PCMA, G722, G723, G726-16, G726-24, G726-32, G726-40, G728, G729, genericAudioCapability, genericAudio (Unique required)

5. **Subname**—Enter a subname to distinguish this genericAudio media profile.
6. Click **Apply**. The genericAudio media profile appears in the Media profile table.
7. Click the genericAudio profile in the Media profile table to choose it and click **Edit**.
8. **Media type**—Choose **video** from the combination box as the media type to use for this media profile.

Media type: audio

control, application, text, audio, data, image, video

9. **Payload type**—Enter the payload type to use for the generic video media profile.

Payload type: 11

10. **Transport protocol**—Select the transport type to use for the generic video media profile from the combination box.

Transport protocol: RTP/AVP

UDP, RTP/AVP

11. Complete the rest of the media profile configuration as needed.
12. Click **Apply**.

H.323 LRQ Alternate Routing

There are networks where the Net-Net SBC is positioned so that it needs to send an H.225 LRQ request to one signaling entity, and then fall back to another signaling entity when there are no resources available on the first. This might be the case when network contain elements that have limited amounts of channels and/or ports.

To handle situations like this one, the Net-Net SBC can be configured for H.323 LRQ alternate routing.

Without this feature enabled, the Net-Net SBC performs H.323 alternate routing for an H.323 call by finding the alternate route for a local policy when the call setup using H.225/Q.931 fails. Some network configurations, however, require that an LRQ message be sent to a gatekeeper prior to call setup in order to request the destination call signaling address—meaning that the Net-Net SBC will release the call if it does not receive an LCF for that LRQ.

With H.323 LRQ alternate routing enabled, the Net-Net SBC can route the call even when it does not receive the LCF.

How It Works

When the Net-Net SBC routes an H.323 call using a local policy and the applicable route specifies gatekeeper/session agent as the next hop, the Net-Net SBC must send that gatekeeper an LRQ to request the destination for the call signaling address. After it sends the LRQ, the Net-Net SBC might receive either an LCF or an LRJ, or it might receive no response at all. Upon failure—either the receipt of an LRJ or no response within a timeout period—the Net-Net SBC tries alternate routes (additional routing policies) until the call is either set up or the routing list ends. For each alternate route, if the next hop is a gatekeeper/session agent, the Net-Net SBC sends an LRQ to the gatekeeper in order to request the destination call signaling address. Otherwise, the Net-Net SBC simply sets up the call.

Session Agent Constraints

If an out of service session agent is part of a route, the Net-Net SBC skips it when using alternate routing and uses other routes for the policy.

A session agent might go out of service when it exceeds the maximum number of consecutive transaction timeouts to the maximum number of allowable transaction timeouts. Applicable session agent parameters of note are:

- **trans-timeouts** session agent option—Maximum number of allowable transaction timeouts (default is 5)
- **No response time to resume** session agent constraint—A session agent out of service until the no-response period expires (or until the Net-Net SBC receives a message from the session agent)
- **In service period** session agent constraint—Amount of time that elapses before a session agent is put back in service after the **No response time to resume** period has passed

Maximum Number of Consecutive LRJs

By default, the Net-Net SBC continues to send LRQ messages to a session agent even if the session agent has already sent an LRJ. However, you might want to place a session agent out of service when it has sent a certain number of LRJs; doing so allows alternate routing to take place faster, but this is an optional feature.

To configure an LRJ threshold, you add the **max-lrj** value to an H.323 session agent's **options** parameter; instructions for how to set it and the required syntax appear

below. If you do not set this option, then the Net-Net SBC will not put session agents out of service for matters related to LRJs.

If you do set this option (to a non-zero value), then the Net-Net SBC keeps a count of the LRJs received from a session agent. When it receives an LCF from a session agent, the Net-Net SBC resets the counter to zero. This count is used internally only and is not accessible through statistics displays.

If a session agent exceeds the maximum number of LRJs and goes out of service, it remains in that state until the **No response time to resume** period has passed and it has transitioned through the **In service period** time. If the **No response time to resume** period is zero, then the session agent is never put out of service.

Caveats

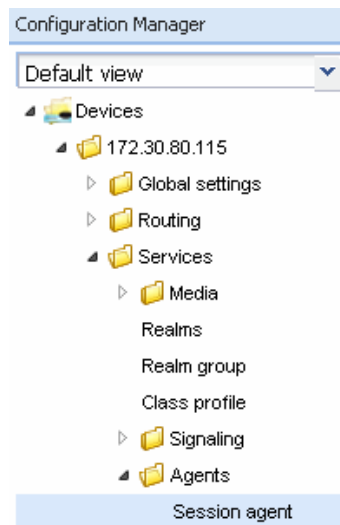
The Net-Net SBC does not support H.323 LRQ alternate routing for these scenarios:

- Calls that require translation between SIP and H.323 (IWF calls)
- For pure H.323 calls where the ingress H.323 interface (stack) is associated with another H.323 interface (stack) that has a valid gatekeeper defined; if there is no valid gatekeeper for the egress interface (stack), this feature may apply

Limiting the Number of Consecutive LRJs

To limit the number of LRJs received from an H.323 session agent before putting it out of service:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select an H.323 session agent in the table to select it and click **Edit**. The Edit Session agent window appears.
4. Click **Options**. The Options dialog box appears.
5. Click **max-lrj** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
7. Click **OK**.
8. Click **Apply**.

H.323 Source Call Signaling Port

The Net-Net SBC now supports a new H.323 interface (stack) option that enables using the Q.931 port value for the port field in the sourceCallSignalAddress parameter in an H.225 Setup message.

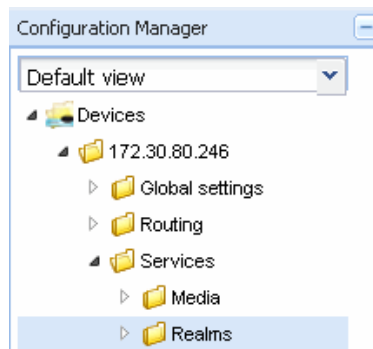
This enhancement is useful for customers who configure a separate H.323 interface (stack) on the core side for each external IP-PBX.

Net-Net Central Instructions and Examples

You configure this feature on a per-interface/stack basis.

To enable the H.323 source call signaling port feature:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab.
7. Click **Options**. The Options dialog box appears.
8. Click **srcCallSignallingPort** in the Available options list to select it.
9. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
10. Click **OK**.
11. Click **Apply**.

H.323 Switchover Behavior

The Net-Net SBC replicates a portion of H.323 signaling/state information across its two SPUs. When an SPU switchover occurs, calls are maintained, however the Net-Net SBC begins a process of gracefully terminating them. Likewise, respective STOP CDRs are generated in order to keep track of the calls' terminated status.

STOP CDRs

The STOP CDR generated after an SPU switchover uses a combination of the *Acme-Disconnect-Cause* and *Acct-Terminate-Cause* attributes to uniquely identify an SPU

switchover as the reason for the dropped call. The values for these attributes appear as follows:

- Acct-Terminate-Cause = 6 (Admin Reset)
- Acme-Disconnect-Cause = 3 (Internal Disconnect)

Graceful Media Stream Termination

Dropping a large number of media streams at one time can place a heavy load on the Net-Net SBC's processing capability and can have a negative impact on call processing. In order to reduce the impact on call processing, media streams will be dropped in a graceful, controlled manner.

Every 10 seconds, the Net-Net SBC drops a fixed number of media streams associated with calls that are active at the time of SPU switchover. The number of streams dropped at the end of each interval is configurable with the **Maximum number of media streams dropped per time interval** parameter. In addition, the CPU load threshold at which to scale back the drop rate is configurable with the **CPU utilization allowed before media stream dropping is throttled** parameter. Finally, the high CPU load media stream drop rate is configured with the **Maximum number of media streams dropped per time interval when CPU limit is exceeded** parameter.

H.323 Switchover Configuration

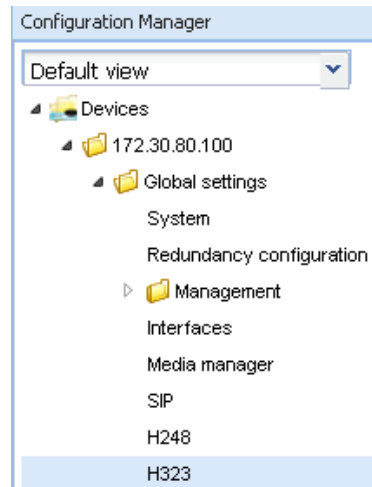
You can configure the Net-Net SBC to perform one of the following four actions after an SPU switchover

- none—switchover, media for calls continues until the flow guard timers expire
- drop media—switchover and drop H.323 calls gracefully
- stop cdr—switchover, do not force any calls to terminate, send STOP CDRs when calls are terminated by UAs
- drop stop—switchover, gracefully terminate media streams, send respective STOP CDRs

Configuring H.323 Calls Following SPU Switchover

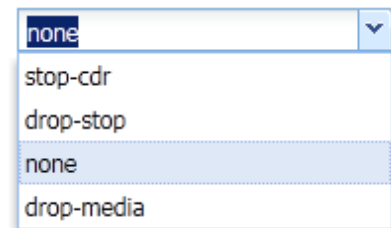
To configure how H.323 calls are dropped after an SPU switchover:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Scroll to H.323 SPU switchover.
5. **Mode**—Click one of the following values in the drop-down list for the SPU switchover mode for H.323 calls. The default is **none**. The valid values are:
 - **none**—switchover, immediately terminating all H.323 calls
 - **drop-media**—switchover and drop H.323 calls gracefully
 - **stop-cdr**—switchover, do not force any calls to terminate, send STOP CDRs when calls are terminated by UAs
 - **drop-stop**—switchover, gracefully terminate media streams, send respective STOP CDRs

Mode



6. **CPU utilization allowed before media stream dropping is throttled**—Enter the maximum CPU load before media streams are dropped at the lower, red-load-max-media value. The default is 30. Valid values range from 1 - 99.

CPU utilization allowed before media stream dropping is throttled

30

7. **Maximum number of media streams dropped per time interval**—Enter the maximum number of media streams to drop during a 10-second drop interval. The default is 15. Valid values range from 0 - 100.

Maximum number of media streams dropped per time interval

15

8. **Maximum number of media streams dropped per time interval when CPU limit is exceeded**—Enter the number of media streams to drop during a 10-second drop interval after the **CPU utilization allowed before media stream dropping is throttled** has been exceeded. The default is 3. Valid values range from 0 - 100.

Maximum number of media streams
dropped per time interval when CPU limit is
exceeded

9. Click **Apply**.

ENUM/Local Route Table Support for H.323 and IWF

Local Route Table (LRT) support for H.323 and IWF is compatible with that currently offered for SIP. LRT and ENUM provide the Net-Net SBC with the ability to perform routing based on ENUM queries to a DNS server or local to an onboard database.

For the LRT feature, this means that entries in the local routing table now include those prefixed with the **h323:** URI scheme, indicating that H.323 is the next hop protocol.

IWF Considerations

When the system performs a local policy lookup for an incoming SIP or H.323 call and determines an ENUM/LRT server is the next hop, it queries that ENUM/LRT server. The response will include the URI scheme, indicating the next hop protocol and the hostname/IP address representing the next hop. For cases where the incoming call signaling protocol and the URI scheme of the ENUM/LRT response are the same, the call requires no interworking. The Net-Net SBC can simply route the egress call leg to the specified next hop.

However, interworking is required when the incoming signaling protocol and the URI scheme of the ENUM/LRT response do not match. In these cases, the Net-Net SBC interworks between SIP and H.323, routing the call to the appropriate next hop.

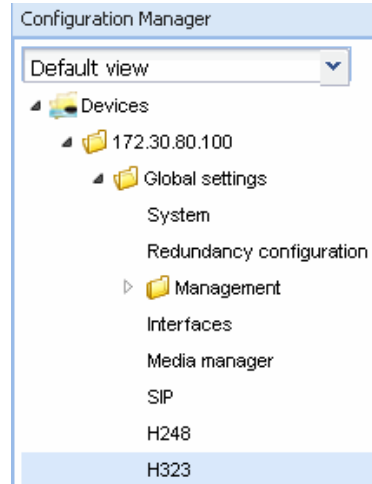
In addition, the Net-Net SBC compares the URI scheme returned in the ENUM/LRT response to the application protocol specified in the policy attributes. If the URI scheme is SIP but the policy attributes indicate H.323, the route is deemed invalid. The same is true for an H.323 URI scheme and SIP route.

Configuring LRT Support

In order for LRT to work for H.323 and IWF calls, you do not have to perform any special configuration. However, you can configure the system to match ENUM/LRT responses against session agent groups, and then use those SAGs for routing.

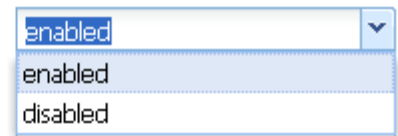
To enable matching ENUM/Local Route Table responses for H.323 SAG routing:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. **ENUM SAG match**—Select enabled from the combination box to allow the Net-Net SBC to perform matching against the hostnames in ENUM/LRT lookup responses and session agent groups. If there is a match, the Net-Net SBC uses the matching SAG for routing. If no match is found, normal ENUM/LRT routing proceeds.

ENUM SAG match



5. Click **Apply**.

H.323 Switchover Behavior

The Net-Net SBC replicates a portion of H.323 signaling/state information across its two SPUs. When an SPU switchover occurs, calls are maintained, however the Net-Net SBC begins a process of gracefully terminating them. Likewise, respective STOP CDRs are generated in order to keep track of the calls' terminated status.

STOP CDRs

The STOP CDR generated after an SPU switchover uses a combination of the *Acme-Disconnect-Cause* and *Acct-Terminate-Cause* attributes to uniquely identify an SPU switchover as the reason for the dropped call. The values for these attributes appear as follows:

- Acct-Terminate-Cause = 6 (Admin Reset)
- Acme-Disconnect-Cause = 3 (Internal Disconnect)

Graceful Media Stream Termination

Dropping a large number of media streams at one time can place a heavy load on the Net-Net SBC's processing capability and can have a negative impact on call processing. In order to reduce the impact on call processing, media streams will be dropped in a graceful, controlled manner.

Every 10 seconds, the Net-Net SBC drops a fixed number of media streams associated with calls that are active at the time of SPU switchover. The number of streams dropped at the end of each interval is configurable with the **Maximum number of media streams dropped per time interval** parameter. In addition, the CPU load threshold at which to scale back the drop rate is configurable with the **CPU utilization allowed before media stream dropping is throttled** parameter. Finally, the high CPU load media stream drop rate is configured with the **Maximum number of media streams dropped per time interval when CPU limit is exceeded** parameter.

H.323 Switchover Configuration

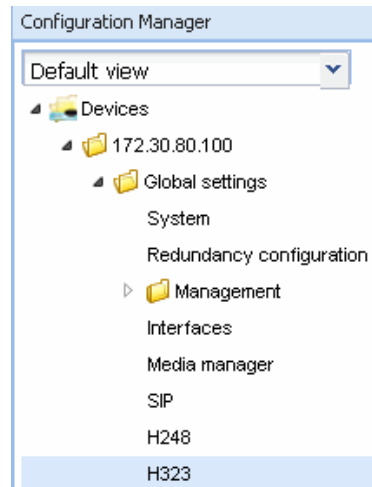
You can configure the Net-Net SBC to perform one of the following four actions after an SPU switchover:

- none—switchover, media for calls continues until the flow guard timers expire
- drop-media—switchover and drop H.323 calls gracefully
- stop-cdr—switchover, do not force any calls to terminate, send STOP CDRs when calls are terminated by UAs
- drop-stop—switchover, gracefully terminate media streams, send respective STOP CDRs

Configuring H.323 Calls Following SPU Switchover

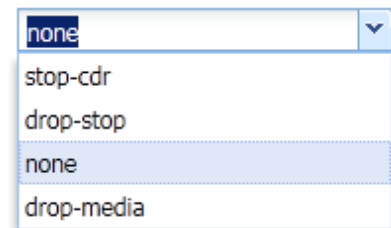
To configure how H.323 calls are dropped after an SPU switchover:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Scroll to H.323 SPU switchover.
5. **Mode**—Click one of the following values in the drop-down list for the SPU switchover mode for H.323 calls. The default is **none**. The valid values are:
 - **none**—switchover, immediately terminating all H.323 calls
 - **drop-media**—switchover and drop H.323 calls gracefully
 - **stop-cdr**—switchover, do not force any calls to terminate, send STOP CDRs when calls are terminated by UAs
 - **drop-stop**—switchover, gracefully terminate media streams, send respective STOP CDRs

Mode



6. **CPU utilization allowed before media stream dropping is throttled**—Enter the maximum CPU load before media streams are dropped at the lower, red-load-max-media value. The default is 30. Valid values range from 1 - 99.

CPU utilization allowed before media stream dropping is throttled

30

7. **Maximum number of media streams dropped per time interval**—Enter the maximum number of media streams to drop during a 10-second drop interval. The default is 15. Valid values range from 0 - 100.

Maximum number of media streams dropped per time interval

15

8. **Maximum number of media streams dropped per time interval when CPU limit is exceeded**—Enter the number of media streams to drop during a 10-second drop interval after the **CPU utilization allowed before media stream dropping is throttled** has been exceeded. The default is 3. Valid values range from 0 - 100.

Maximum number of media streams
dropped per time interval when CPU limit is
exceeded

9. Click **Apply**.

H.323 - G726 Transcoding

H.323 - G726 Transcoding

You can now configure your D-Series SBC to support H.323 - G726 transcoding.

International Peering with IWF and H.323 Calls

In previous Net-Net SBC releases, SIP to H.323 IWF calls default to a National Q.931 Number Type and it is not possible to change it to an International number. This feature allows you to override that behavior by configuring the option **cpnType=X**, where X is an integer that maps to various Q.931 Number Types. When this option is set, Q.931 Number Type for both calling party and called party are updated to the configured value for all outgoing calls on the h323-stack.

In addition to SIP to H.323 IWF calls, the **cpnType=X** option is used to set the Q.931 Number Type for H.323-H.323 calls.

The following is a list of possible values for X:

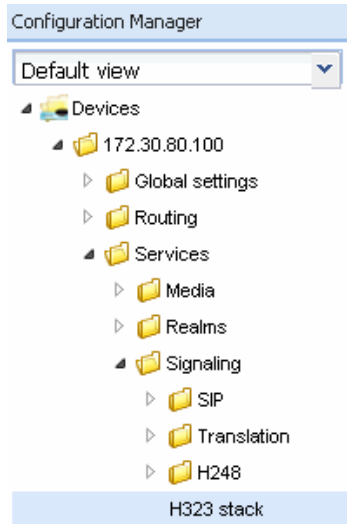
- 0—Unknown public number
- 1—International public number
- 2—National public number
- 3—Specific public network number
- 4—Public subscriber number
- 5—Public abbreviated number
- 6—Private abbreviated number

Configure International Peering for H.323 Calls

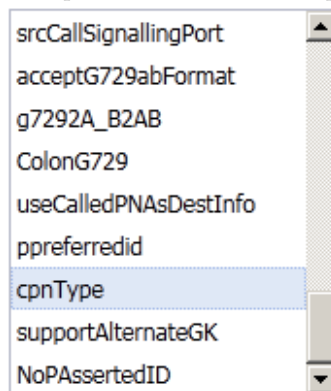
You configure this feature as an option in the **h323-stack** configuration.

To configure the `cpnType=X` option for H323-H323 calls:

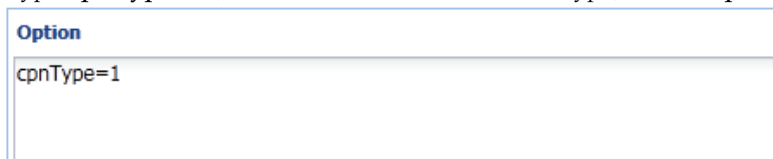
1. Expand the Configuration Manager slider > Services > Signaling > H323 Stack.



2. Click H323 Stack.
3. Click the H.323 stack you want to configure and click **Edit**. The H323 stack configuration window appears.
4. Click on the **H.323 service mode** tab and scroll to the Options section of the content area.
5. Click **Options**. The Options window appears.
6. Select **cpnType** in the Available Options list to select it and click  to move the option to the Selected Options list.



7. Type **cpnType=x** (where x is the Q.931 Number Type) in the **Option** field.



8. Click **OK**.
9. Click **Apply**.

ENUM for H.323 and SIP-H.323 IWF

H.323 and SIP-H.323 IWF session can take full advantage of the Net-Net SBC's ENUM support. There is not configuration required. ENUM support works automatically for all H.323-originated sessions.

There is no configuration required for this feature. It is enabled automatically.

H.323 Stack Monitoring

The H.323 stack/interface configuration provides a way to set one of three alarm thresholds on a per-stack basis, based on the max-calls value. Upon a threshold being crossed the Net-Net SBC will trigger an alarm and an SNMP trap.

To prevent the alarm from firing continuously as call volume through the stack varies, each severity level has an has a reset value below the TCA you set. In addition, each threshold value resets when:

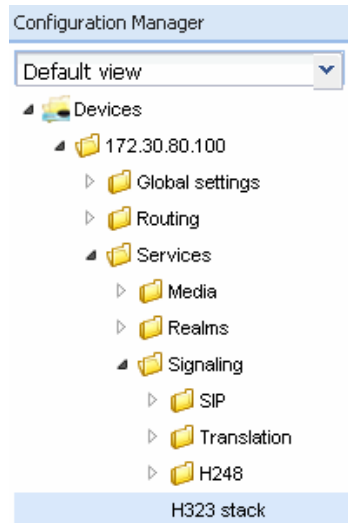
- An alarm with a higher severity is triggered, or
- The built-in reset value for the threshold level is 1% less than the parameter value

Configure H.323 Stack Monitoring

This section shows you how to configure H.323 stack monitoring for one H.323 stack configuration. This example shows one instance of the alarm-threshold sub-configuration being established; remember that you can set three—critical, major, and minor. Simply repeat the configuration steps to add more severity levels.

To set up H.323 stack monitoring:

1. Expand the Configuration Manager slider > Services > Signaling > H323 Stack.



2. Click H323 Stack.
3. Click the H323 Stack you want to configure and click **Edit**. The H323 Stack Configuration window appears.
4. Scroll to the Alarm threshold table and click **Add**.

5. **Severity**—Enter the type of severity level for the alarm you want to define. Choose from: **critical**, **major**, or **minor**. This value is required, and defaults to **minor**.

*Severity

6. **Value**—Enter the percentage of the number of calls defined in the **max-calls** parameter that triggers the alarm. For example, if you want to set a minor alarm to fire when the call rate through the stack reaches half the **max-calls** value, enter **50** (meaning 50%). The default value for this parameter is 0, which disables the alarm.

*Value (%)

Remember that if the number of calls falls to below 1% of the max-calls threshold you set, the clear trap fires.

7. Click **Apply**.

Cause Code Completeness

H.323 CDR Generating Behaviors

By enabling enhanced CDR in the H.323 Config, the following cause codes will be reported for the identified action.

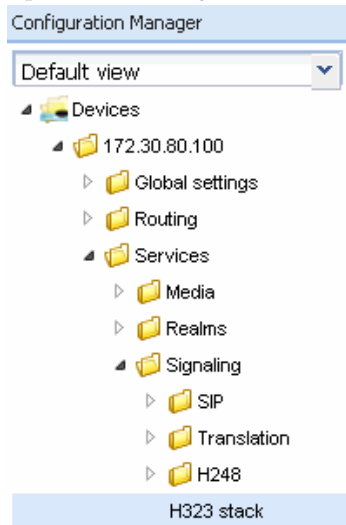
- The Net-Net SBC may reject a call because the applicable transcoding policy does not support the codecs offered by the remote H.323 endpoint. In such a case, when the **enhanced-cdr** option is configured in the H.323 config, the Net-Net SBC will insert cause code 44 in both the h323-disconnect-cause and Acme-disconnect-cause in the CDR. In addition, the disconnect reason remains blank. When **enhanced-cdr** is not configured, h323-disconnect-cause and Acme-disconnect-cause have a value of 3 in the CDR and the “No Route to Destination” disconnect reason is sent to the endpoint.
- The Net-Net SBC may reject a call because the previous hop is not authorized, due to allow-anonymous settings. In such a case, when the **enhanced-cdr** option is configured in the H.323 config, the Net-Net SBC will insert cause code 21 in both the h323-disconnect-cause and Acme-disconnect-cause in the CDR. In addition, a disconnect reason of “SecurityDenied” will be sent to the caller. When **enhanced-cdr** is not configured, h323-disconnect-cause and Acme-disconnect-cause are left blank in the CDR.
- The Net-Net SBC may reject a call because it is out of bandwidth for the codecs offered by the remote H.323 endpoint. In such a case, when the **enhanced-cdr** option is configured in the H.323 config, the Net-Net SBC will insert cause code 34 in both the h323-disconnect-cause and Acme-disconnect-cause in the CDR. In addition, a blank disconnect reason will be sent to the caller. When **enhanced-cdr** is not configured, h323-disconnect-cause and Acme-disconnect-cause are left blank in the CDR.
- The Net-Net SBC may reject a call because of outbound session constraints being exceeded defined in either the Session Agent or Realm configurations. In such a case when the **enhanced-cdr** option is configured in the H.323 config, the Net-Net SBC will insert cause code 34 in both the h323-disconnect-cause and Acme-disconnect-cause in the CDR. In addition, a blank disconnect reason will be sent to the caller.
- When ingress number translation is applied to an H.323 call, and the **enhanced-cdr** option is configured in the H.323 config:
 - The Calling-Station-Id VSA contains the original calling party’s number.

- The Called-Station-Id VSA contains the original called party's number.
- The Acme-Primary-Routing-Number VSA contains the called party's number after the inbound translation, the one that is used to do the local policy lookup.
- The Acme-Egress-Final-Routing-Number VSA contains the called party's number after the outbound translation.
- When the Net-Net SBC clears a call after a media guard timer expires, and the **enhanced-cdr** option is configured in the H.323 config:
 - The Acme-Disconnect-Cause VSA value is set to 16.
 - The Acme-Disconnect-Initiator VSA value is set to 3.
 - The H323-Disconnect-Cause VSA value is set to 6.
- When the **enhanced-cdr** option is configured in the H.323 config, and the outbound leg of a call is rejected with an LRJ/ARJ, the `rasReason` of `undefinedReason` will be mapped to the `releaseCompleteReason` `undefinedReason`. Also, the Net-Net SBC will insert cause code 31 in both the `h323-disconnect-cause` and `Acme-disconnect-cause` in the CDR.

Enable H.323 CDR Handling

To enable CDR handling of the above H.323 scenarios:

1. Expand the Configuration Manager slider > Services > Signaling > H323 Stack.



2. Click H323 Stack.
3. Select the H323 stack you wish to use and click **Edit**.
4. Scroll to Options for H323 Service and click **Options**. The Options window appears.
5. Select **enhanced-cdr** in the Available options list to select it and click  to move the option to the Selected options list.
6. Click **OK**.
7. Click **Apply**.

H.323 Generating Behavior for OOS Session Agents

When the Net-Net SBC fails a call because an applicable H.323 session agent goes out of service, you can specify a custom cause code to write to the CDR for the failed call.

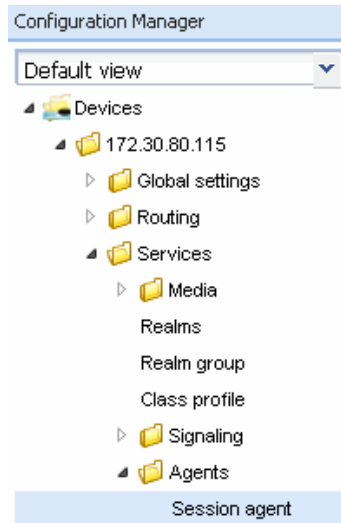
When the option **sa-oos-code=x** is configured in the session agent configuration element, the user-defined cause code of x written to the h323-disconnect-cause and Acme-disconnect-cause fields in the CDR. In addition, the user-defined value is sent as the the Q.850 cause code to the H.323 endpoint.

If the next-hop for an H.323 Session is a H.323 session agent which does not respond to the SETUP request, the call will be rejected with a cause code of 102. After 5 consecutive failures, the session agent is set to OOS. On the 6th failed call attempt to the failed session agent, the call is rejected with a cause code of 34. If the **sa-oos-code** option is configured in this case, the user defined value will be used for the 6th and subsequent failed attempt.

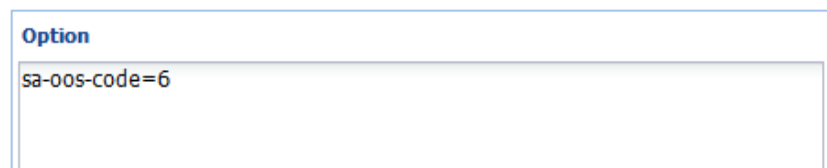
Enable Custom H.323 Cause Code for OOS Session Agents

To enable CDR handling of out of service session agents in the above H.323 scenarios:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the session agent you would like to use and click Edit.
4. Click **Options**. The Options window appears.
5. Select **sa-oos-code** in the Available options list and click  to move the option to the Selected options list.
6. Set the Option parameter by typing the option name **sa-oos-code=x** (where x is the the Q.850 cause code value that will be used by the SBC to use in both the signaling messages and the CDR when the H.323 SA goes OOS.).



7. Click **OK**.
8. Click **Apply**.

SIP CDR Generating Behaviors

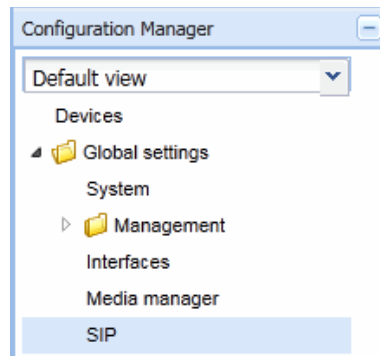
The following accounting behaviors are all enabled at once by setting the **enhanced cdr** option in the SIP config:

- The Net-Net SBC generates a CDR stop record when a call is rejected due to exceeding session constraints. For example: calls rejected with "500 Outbound Constraints Exceeded" generate stop CDRs.
- The Net-Net SBC generates a CDR stop record when a call is rejected due to not finding a route. For example: calls rejected with "404 No Route Found" generate stop CDRs.
- The Net-Net SBC generates a CDR record for all cases where it rejects a well-formed request.

Enable SIP CDR Handling

To enable CDR handling of the above SIP scenario:

1. Expand the Configuration Manager slider -> Global settings.



2. Click SIP. The SIP data window appears in the content area.
3. Scroll to Options for H323 Service and click **Options**. The Options window appears.
4. Select **enhanced-cdr** in the Available options list and click  to move the option to the Selected options list.
5. Click OK.
6. Click Apply.

H.323 Proprietary Identification Mechanism of a Registering Terminal

Certain videoconferencing require endpoints to register with a gatekeeper to join a conference. The endpoints access the gatekeeper over an IP VPN and through a Net-Net SBC, and do not use H.235 authentication. Because the Net-Net SBC hides the accessing topology from the gatekeeper, a malicious endpoint can spoof the identity of the endpoint. To protect against malicious endpoints you can configure the Net-Net SBC to add a proprietary field to incoming registration messages that are forwarded to the gatekeeper. This field enables the gatekeeper to authorize or reject the registration. When added to an H.323 stack that processes registrations The secureRegistration option allows gatekeepers this ability to process registrations.

Requirements

To support the Proprietary Identification Mechanism, follow these requirements for your ingress and egress H.323 interface (stack) configurations.

Ingress (access) h323-stack

Parameter	Settings
h323-stack> isgateway	disabled
h323-stack>process-registration	enabled

Egress (core) h323-stack

Parameter	Settings
h323-stack> isgateway	enabled
h323-stack> gatekeeper 192.168.140.247:1719	non-zero value applicable to your network
h323-stack> process-registration	disabled
h323-stack> q931-start-port	non-zero value applicable to your network
h323-stack> q931-number-ports	non-zero value applicable to your network

How It Works

Registration

An H.225 registrationRequest (RRQ) message for a full (initial) registration is received by the Net-Net SBC's ingress h323-stack. The egress h323-stack parameter for isgateway is enabled. The Net-Net SBC inserts a proprietary terminalAlias in the outgoing registrationRequest message sent to the gatekeeper according to the following scheme; in order.

1. If any terminalAlias fields with AliasAddress of type email-ID are contained in the incoming RRQ, the Net-Net SBC will remove them in the outgoing RRQ.
2. The Net-Net SBC constructs the terminalAlias to be inserted in the outgoing RRQ is constructed by the Net-Net SBC as an AliasAddress of type email-ID, which is composed of a concatenation of dialledDigits, vlanID of the ingress h323-stack, and the IP address of the egress h323-stack as follows:
 - 2a. dialledDigits: The phone number announced by the terminal in the incoming RRQ's terminalAlias list. If several aliases of type dialledDigits are present, the Net-Net SBC uses the last dialledDigits alias in the list. If there is no alias of type dialledDigits present in the incoming alias list, the inserted dialledDigits field in the constructed email-ID will be empty.
 - 2b. vlanid: This value corresponds to the physical VLAN number (sub-port-id) of the network-interface on which the incoming RRQ message was received.
 - 2c. egressH323StackIP: This value corresponds to the local-ip address of the egress h323-stack.

Format of email-ID terminal Alias (ABNF language)

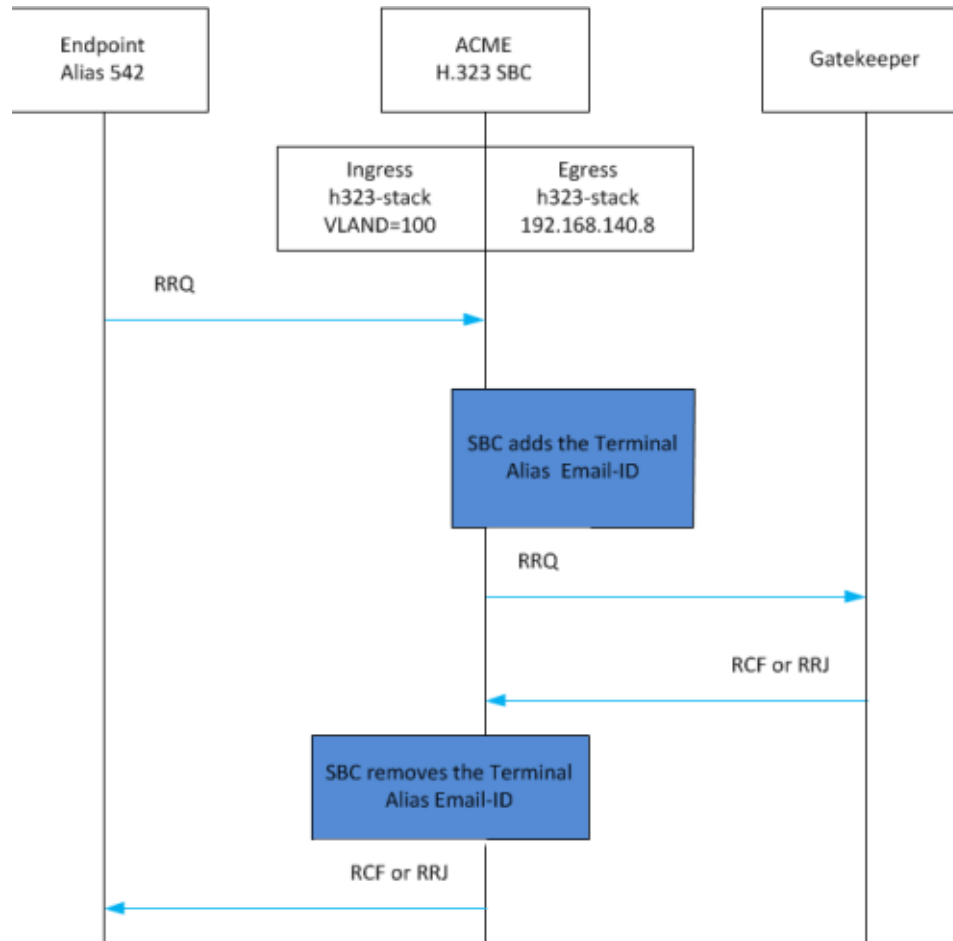
email-ID	=dialledDigits	"," vlanid ","vlanid@" egressH323StackIP
dialledDigits	=DIGIT	
vlanid	=DIGIT	
egressH323StackIP	=IP4-address	

An example of the inserted email-ID terminalAlias looks like this:

542.100.vlanid@192.168.140.8

The Net-Net SBC removes the terminalAlias email-ID from the registrationConfirm or registrationReject sent by the Net-Net SBC back to the registering endpoint on the ingress (access) side. Any terminalAlias email-ID present in the incoming RRQ and removed by the Net-Net SBC in the outgoing RRQ will not be restored in the RCF or RRJ sent by the Net-Net SBC back to the registering endpoint on the ingress (access) side. If the RRJ contains a registrationRejectReason of duplicateAlias, the sequence of AliasAddresses contained in the duplicateAlias node will be operated

upon, and if the registrationRejectReason is invalidTerminalAliases, then the sequence of terminalAlias AliasAddresses will be operated on.



Unregistration

In the unregistration process, H.225 unregistrationRequest (URQ) message is received by the Net-Net SBC's ingress h323-stack. The Net-Net SBC routes the URQ to the egress stack's gatekeeper. The gatekeeper replies with an unregistrationConfirm (UCF) to confirm the unregistration. The Net-Net SBC forwards the UCF to the registering endpoint on the access side.

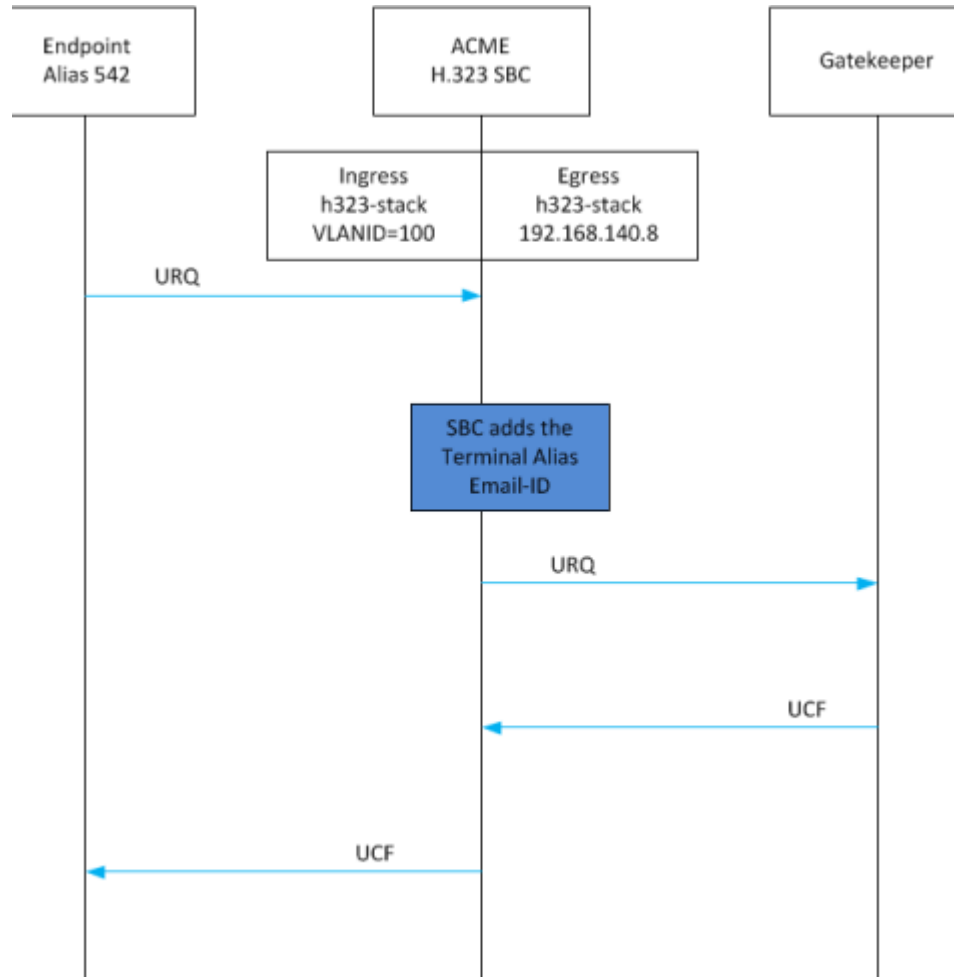
For the incoming (URQ) the h323-stack that processes registrations the Net-Net SBC will insert the outgoing terminalAlias in the outgoing URQ sent to the gatekeeper. The incoming URQ contains all of the terminal aliases (except for the proprietary email-ID) currently registered to the endpoint.

When the incoming URQ contains no terminal aliases (only the endpoint Identifier), the Net-Net SBC will forward the URQ to the gatekeeper without adding the proprietary terminalAlias. When the incoming URQ contains a subset of the aliases currently registered to the endpoint, the Net-Net SBC will forward the URQ to the gatekeeper without adding the proprietary terminalAlias. When a subsequent URQ unregisters the remaining aliases registered to the endpoint, the Net-Net SBC will insert the proprietary terminalAlias in the URQ sent to the gatekeeper. In the unregistration scenario, the unregistrationConfirm and unregistrationReject

messages from the gatekeeper cannot contain any terminalAliases. The Net-Net SBC is not required to remove the email-ID terminalAliases.

The Net-Net SBC will not modify (insert or remove an email-ID terminalAlias) the following RAS messages:

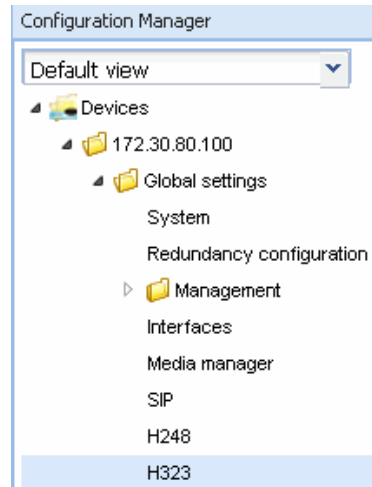
- keepAlive (lightweight) RRQ
- ARQ/ACF/ARJ
- LRQ/LCF/LRJ
- IRR
- UCF



ACLI Instructions and Examples

You use the **secureRegistration** option to process registrations. This option should be set on the h323-stack that is configured to process registrations.

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **secureRegistration** in the Available options list to select it. Click the arrows button to move the option to the Selected options list.
6. Click **OK**.
7. Click **Apply**.

Monitoring

The Net-Net SBC show h323 registrations <reg-key> command shows the registered endpoint when <reg-key>, is equal to the terminalAlias email-ID that was inserted in the outgoing RRQ, and confirmed by the egress h323-stack gatekeeper terminalAlias email-ID, that were present in the incoming RRQ, and removed in the outgoing RRQ will not be shown as a registered alias by the Net-Net SBC.

H.323 Setup without Called Number

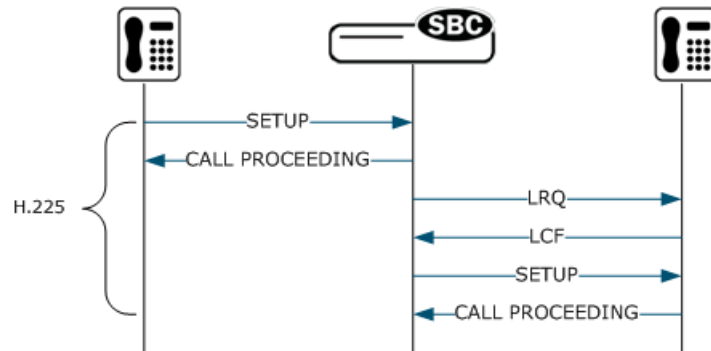
For videoconferencing applications in which Net-Net SBCs are deployed, unregistered endpoints attempt to join conferences by dialing the Net-Net SBC's IP address. The H.225 Setup the Net-Net SBC receives, however, will either:

- Not contain an H.245 destinationAddress field, or
- The H.245 destinationAddress will be present but not configured as an AliasAddress of dialedDigits.

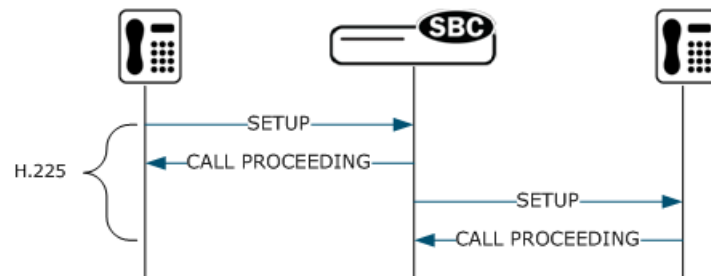
For both formats, the Setup message will not have a Q.931 Called part number, and the H.225 destCallSignalAddress field will contain the ipAddress dialed by the calling party. The lack of the destinationAddress or its failure to be configured as an AliasAddress of dialedDigits keeps the endpoint from joining the conference because no called party number can be determined.

When you configure the insertCallednumber option for an H.323 interface (stack), the Net-Net SBC adds a number you set in the configuration to the appropriate signaling messages. The presence of the number allows the session to proceed.

In Gatekeeper mode, the presence of the `insertCalled` number option instructs the Net-Net SBC to insert a preconfigured number as an `AliasAddress` of `dialedDigits` into the `destinationInfo` field of the LRQ message. In this scenario, the `isgateway` parameter is disabled for the egress H.323 interface, establishing Gatekeeper mode. The Net-Net SBC sends a `locationRequest` to the egress stack's Gatekeeper, which responds with a `locationConfirm`. Upon the receipt of the `locationConfirm`, the Net-Net SBC sends an H.225 Setup to the called party.



In Gateway mode, the presence of the `insertCalled` number option instructs the Net-Net SBC to insert a preconfigured number as an `AliasAddress` of `dialedDigits` into the `destinationAddress` field of Setup message sent to the Gatekeeper. When the `isgateway` parameter is enabled for the egress H.323 interface, Gateway mode is established. The Net-Net SBC routes the call out the egress interface and sends a Setup to the called party.



Caveats

This functionality is not support for sessions interworked between SIP and H.323.

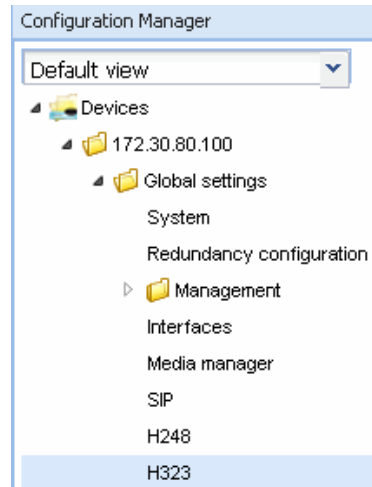
ACLI Instructions and Examples

This section shows you how to configure the `insertCallednumber` option for an H.323 interface. Note that you configure the option itself and the number you want inserted into the relevant message.

When enter the number you want to insert, you can use all numeric character (0, 1, 2, 3, 4, 5, 6, 7, 8, and 9), as well as the askterisk (*), pound sign (#), and comma (.). The value must be between 1-128 characters.

To enable H.323 Setup without Called Number:

1. Expand the Configuration Manager slider > Global settings.

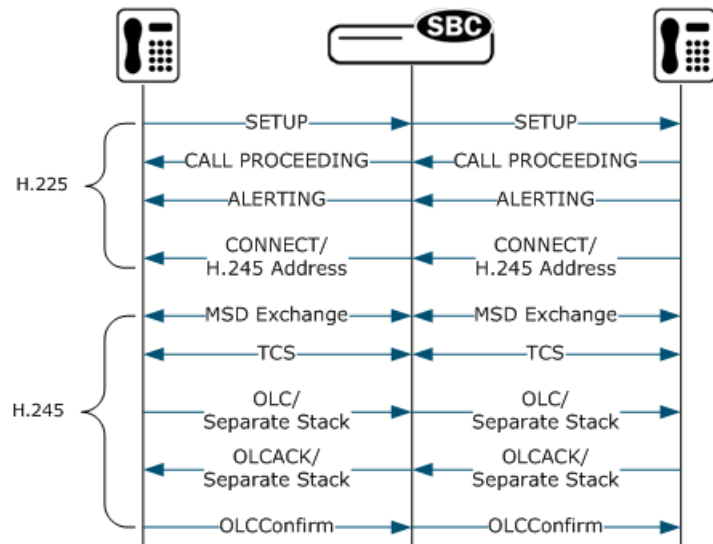


2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **insertCallednumber=x** (where X is number you want inserted into the appropriate message) in the Available options list to select it. Click the arrows button to move the option to the Selected options list.
6. Click **OK**.
7. Click **Apply**.

H.245 Separate Stack

To provide management of its conference services, videoconferencing applications establish a TCP/XML channel negotiated via H.245. To set up the channel, H.245 signaling message use the `separateStack` parameters in the `OpenLogicalChannel` and `OpenLogicalChannelAck` messages in order to exchange IP address and port information. In architectures where it is deployed, the Net-Net SBC must remain transparent during the exchange of H.245 messages that establish that channel.

You can enable this transparency by setting the `separateStack` option in your H.323 global configuration. Configuring this option allows the Net-Net SBC to pass through the H.245 `separateStack` value and its subfields in H.245 `OpenLogicalChannel` and `OpenLogicalChannelAck` messages. To maintain transparency, when you configure this option, the Net-Net SBC will not set up media flows (as it otherwise would) for the IP address and port defined by the `separateStack` transport address.



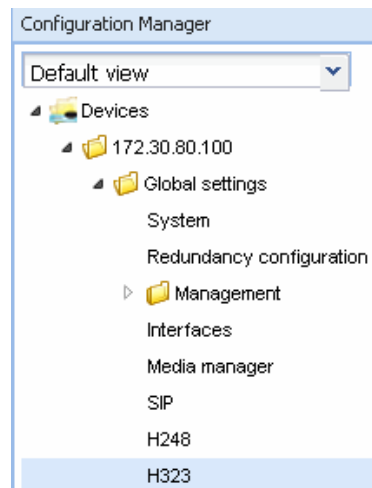
Caveats

- This functionality is not support for sessions interworked between SIP and H.323.
- This functionality does not support H.323 fast start, i.e. H.323 Fast Connect.

ACLI Instructions and Examples

To enable H.245 separate stack support:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **separate stack** in the Available options list to select it. Click the arrows button to move the option to the Selected options list.
6. Click **OK**.

7. Click **Apply**.

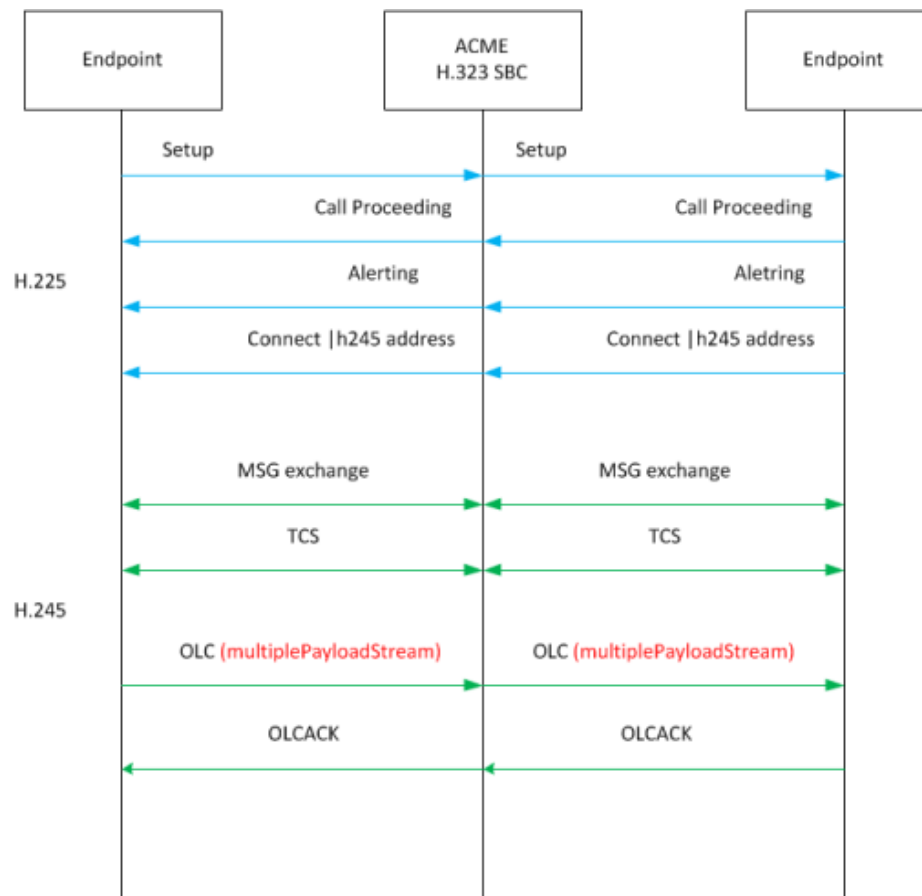
H.264 FEC

To provide error resiliency upon detection of packet loss on media streams the Net-Net SBC uses the Forward Error Correction (FEC). This mechanism protects a media stream by encoding FEC packets within RTP packets. The FEC packets are sent in a different stream than the media it is protecting and uses a different payload type. The FEC packets are encoded within RTP packets and sent over the same IP and port as the media it protects.

The Net-Net SBC transparently passes through FEC-related capabilities in the H.245 TerminalCapabilitySet (TCS) messages. Specifically, this includes the fecCapability capability in the capabilityTable. The Net-Net SBC transparently passes through FEC-related data types in the H.245 OpenLogicalChannel messages. Specifically, this includes the multiplePayloadStream dataType and its subfields. The Net-Net SBC set's up media flows for the FEC channels according to the IP address and port (network and tsapIdentifier) specified in the h2250LogicalChannelParameters field of the H.245 OpenLogicalChannel messages and OpenLogicalChannelAck.

Caveats

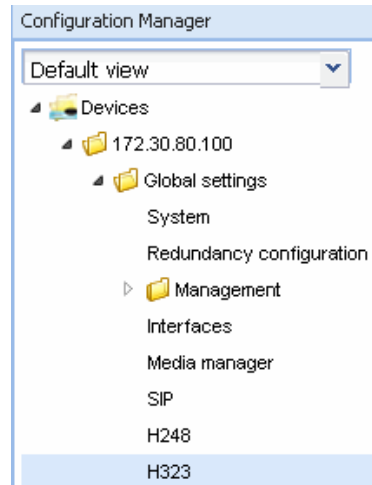
- Only H.323 slowstart to H.323 slowstart call signaling flows are supported.
- SIP H.323.IWF calls are not supported.
- Video is supported all other data types will be rejected.



ACLI Instructions and Examples

To enable the FEC option used to process registrations. Use the (FEC) option on the h323interface (stack) to process registrations.

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **FEC** in the Available options list to select it. Click the arrows button to move the option to the Selected options list.
6. Click **OK**.
7. Click **Apply**.

Introduction

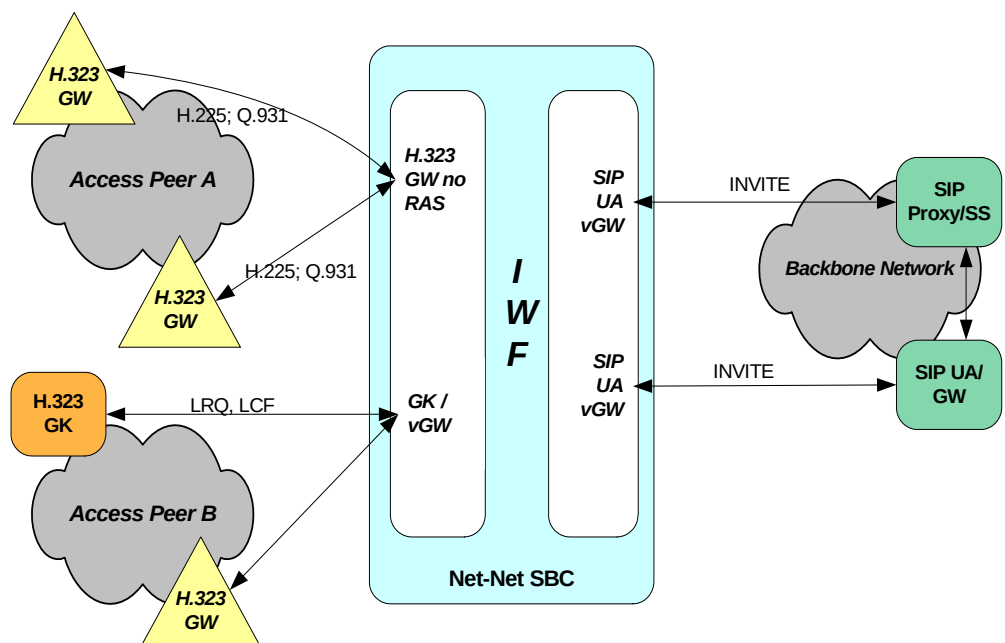
Using the Net-Net SBC's interworking (IWF) function, you can interconnect SIP networks with H.323 networks. Considering the large amount of H.323 deployments already in place and the continuing emergence of SIP in new VoIP deployments, the IWF provides a much-needed solution. SIP providers can maintain a single-protocol backbone while exchanging VoIP sessions with H.323 providers.

The H.323 Signaling Services chapter contains information about the H.323 signaling modes of operation that the Net-Net SBC supports. The following H.323 signaling modes of operation can be used when you use the Net-Net SBC's IWF in an access or a peering solution.

- Back-to-back gateway signaling
- Interworking gatekeeper/gateway

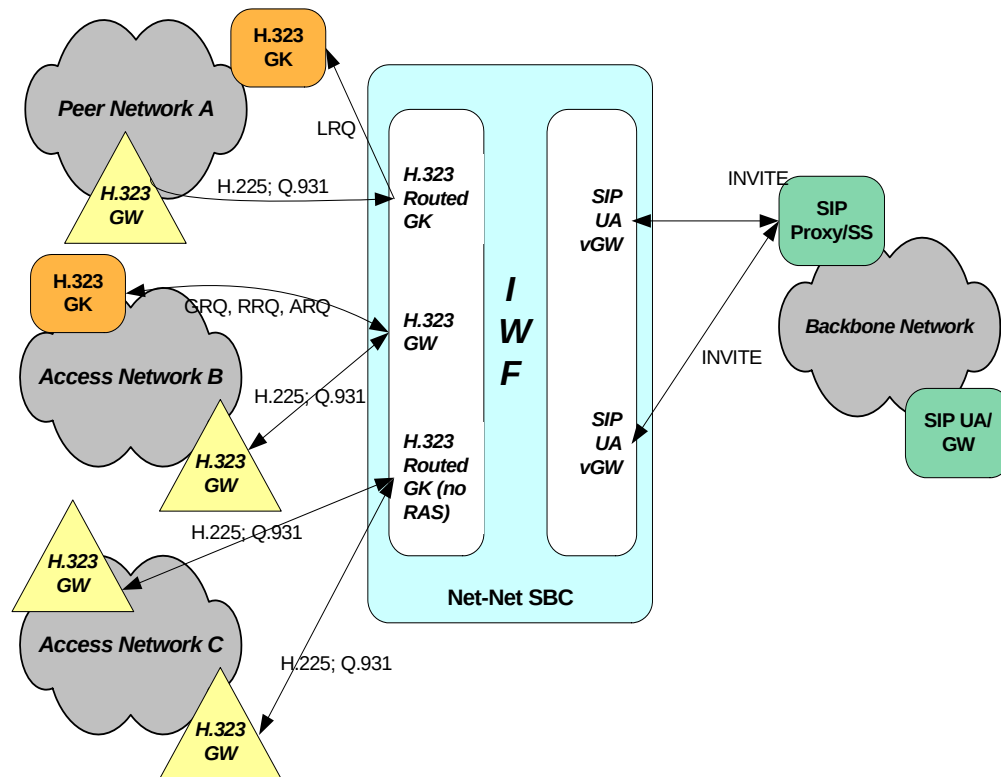
Access Network Application

You can configure your Net-Net SBC so that it provides an access solution for your network. The access solution allows SIP-based hosted communications platforms to be extended to enterprise-based H.323 systems. In the figure below, you can see different types of H.323 signaling modes being interworked with SIP. On the H.323 side, the Net-Net SBC can appear to be a gatekeeper or a gateway, depending on how you configure the H.323 interface. On the SIP side, the Net-Net SBC can appear to be a SIP UA or behave as a virtual gateway.



Networking Peering Application

In the IWF network peering solution, you can see the same network elements at work. However, the H.323 side of this IWF application shows the use of a gatekeeper controlled gateway for Peer Network B. Because this is a peering solution, the SIP side of the Net-Net SBC communicates with the SIP proxy or softswitch in the backbone network rather than with the SIP UA or SIP gateway.



How It Works

The Net-Net SBC supports interworking between SIP and H.323 for H.323 Slow Start and Fast Start calls. In addition to describing IWF sessions when initiated from the H.323 side and from the SIP side (with sample call flows), this section provides information you will need when you configure SIP and H.323.

SIP/H.323 Negotiation: H.323 Fast Start

The Net-Net SBC can perform protocol translations for SIP and H.323 Fast Start, where media capabilities are sent with the Setup request for an H.323 session.

This section's call flow diagrams show how SIP and H.323 messages flow between SIP and H.323 devices, with the Net-Net SBC positioned between the two entities so it can perform translations. The following two sample scenarios with Fast Start appear in the diagrams below, although other scenarios are possible:

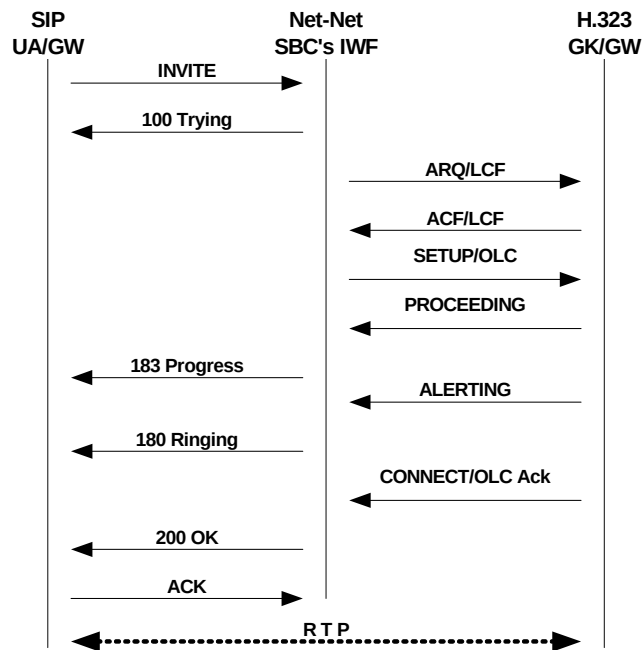
- Calls originating in SIP being translated to H.323 Fast Start
- Calls originating in H.323 Fast Start translated to SIP

SIP to Fast Start H.323

In the following diagram below, a SIP endpoint (such as a UA or a SIP Gateway) initiates a session by sending an INVITE message destined for an H.323 endpoint (a

GK or GW). Between these entities, the Net-Net SBC is positioned to perform interworking. The Net-Net SBC recognizes that the INVITE message is destined for an H.323 device, and returns a 100 Trying message to the SIP endpoint as it attempts to negotiate the H.323 side of the session. This negotiation starts when the Net-Net SBC initiates the RAS process with the H.323 endpoint by sending either an ARQ or an LRQ, allowing the Net-Net SBC to determine if the H.323 endpoint will accept the session.

Once the H.323 endpoint responds with an ACF or LCF, the Net-Net SBC reissues the SIP INVITE on the H.323 side as an H.225 Setup, which is sent with the OLC. Then the H.323 endpoint responds with Proceeding and Alerting messages (which correspond respectively to SIP 183 Progress and 180 Ringing messages). At that point, the H.323 endpoint sends a Connect message that includes the OpenLogicalChannel message (OLC), announcing the logical channel for media flows has been set up. The Net-Net SBC converts the H.323 OLC to a SIP 200 OK. After receiving the 200 OK, the SIP endpoint sends an ACK, confirming that the session has been established. Because there is no H.323 equivalent for the SIP ACK, the Net-Net SBC does not generate a corresponding message on the H.323 side. At this point, the session is fully established and RTP flows between the endpoints.

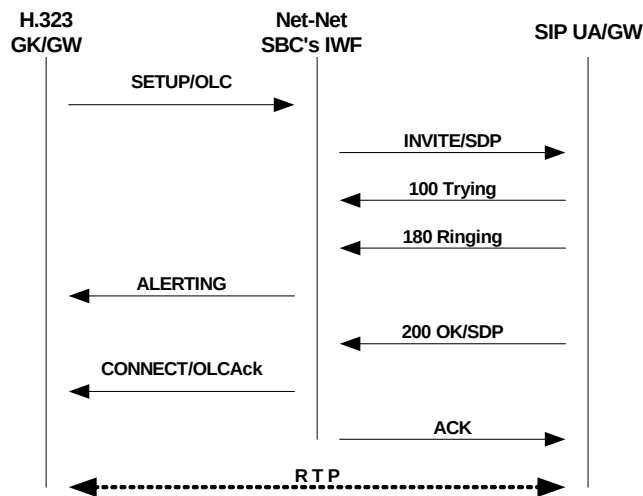


H.323 Fast Start to SIP

In the diagram below, an H.323 endpoint (a GK or GW) initiates a session by sending a Setup request destined for a SIP endpoint (such as a UA or a SIP Gateway). Between these entities, the Net-Net SBC is positioned to perform interworking. The H.323 endpoint has completed the RAS process prior to sending the SETUP message.

The Net-Net SBC receives the Setup message and then sends a SIP INVITE on the SIP side. The SIP endpoint responds with a 100 Trying; the Net-Net SBC does not resend this message on the H.323 side. Next, the SIP endpoint issues a 180 Ringing message, which the Net-Net SBC reissues to the H.323 endpoint as an Alerting message. The SIP endpoint then sends a 200 OK, retransmitted by the Net-Net SBC

as a Connect message that includes an OLC. Once the Net-Net SBC sends an ACK to the SIP endpoint, RTP flows between the endpoints.



SIP/H.323 Negotiation: H.323 Slow Start

The Net-Net SBC can also perform protocol translations for SIP and H.323 Slow Start, where—unlike the cases with Fast Start described above—media information is not sent with the Setup request for an H.323 session. For H.323 Slow Start, media is negotiated after the session is established.

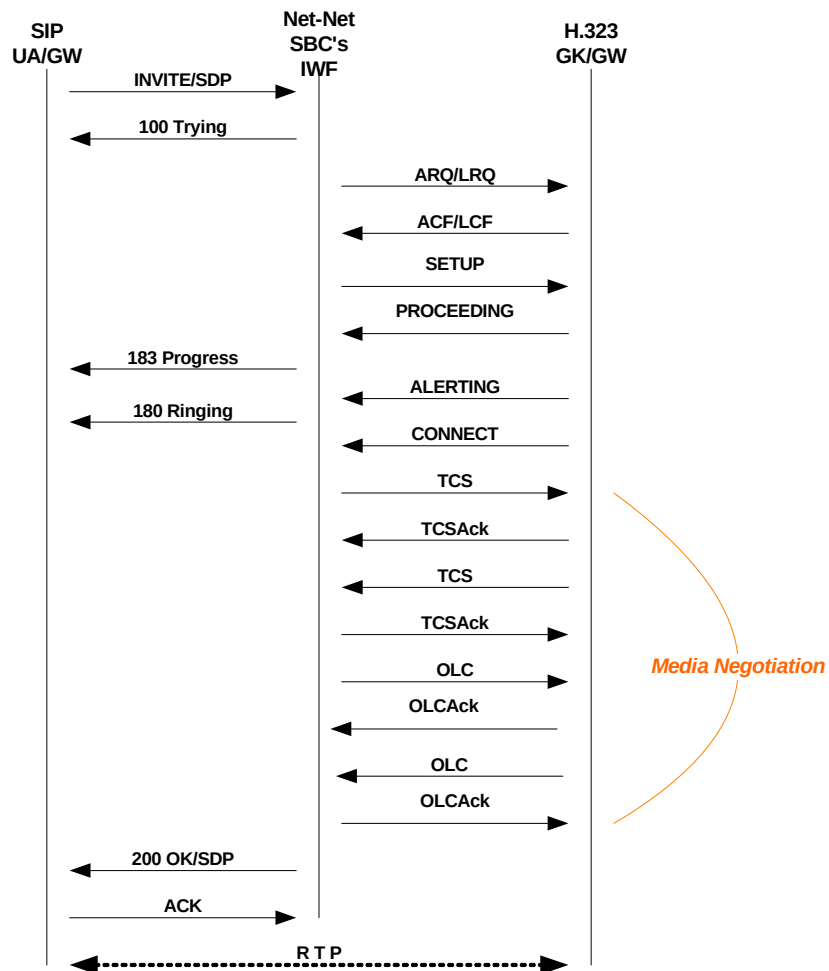
This section's call flow diagrams show how SIP and H.323 messages flow between SIP UA/GW and an H.323 GK/GW, with the Net-Net SBC positioned between the two entities so it can perform translations. Two sample scenarios with Slow Start appear in the diagrams below:

- SIP being interworked to Slow Start H.323
- Slow Start H.323 being interworked to SIP

H.323 SIP to Slow Start

In the following diagram below, a SIP endpoint (such as a UA or a SIP Gateway) initiates a session by sending an INVITE request destined for an H.323 Slow Start endpoint (a GK or GW). Between these entities, the Net-Net SBC is positioned to perform interworking.

The call flow for this type of translation works fundamentally the same way that the translation does for SIP to Fast Start H.323, with the exception of how the media is established. Media is negotiated through the exchange of TCS and OLC messages after the H.323 Connect and SIP 180 Ringing messages have been sent. The first TCS message is sent from the Net-Net SBC to the H.323 endpoint, and it contains information about media capabilities in SDP. The H.323 endpoint accepts and acknowledges this information with a TCS Ack message. Then the H.323 endpoint sends a second TCS, carrying information about the Gateway's capabilities, that the Net-Net SBC accepts and acknowledges. The H.323 endpoint and the Net-Net SBC then exchange OLC and OLC Ack messages that establish the operating mode and Gateway capability. Finally, the Net-Net SBC completes the 200 OK/ACK sequence on the SIP side, and RTP flows between the two endpoints.

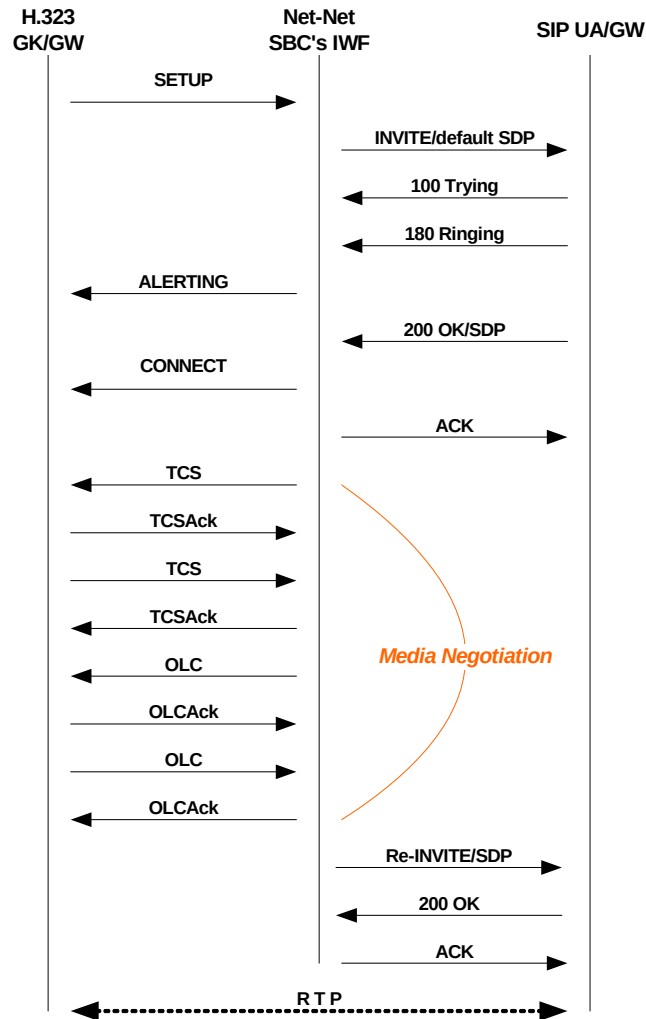


H.323 Slow Start to SIP

In the following diagram below, an H.323 endpoint (GW or GK) initiates a session by sending a Setup request destined for a SIP endpoint (such as a UA or a SIP Gateway). Between these entities, the Net-Net SBC is positioned to perform interworking. The H.323 endpoint has completed the RAS process prior to sending the SETUP message.

The call flow for this type of translation works fundamentally the same way that the translation does for H.323 Fast Start to SIP, with the exception of how the media is established. When the Net-Net SBC receives an H.323 message destined for a SIP endpoint, it sends a SIP INVITE message that includes default SDP to that SIP endpoint. The default SDP is constructed using information in the media profiles listed for the IWF configuration; if necessary, this media information is amended later in the sequence. Once the call is set up, the Net-Net SBC negotiates media with the H.323 endpoint through a series of TCS/TCS Ack and OLC/OLC Ack messages that establish the operating mode and Gateway capability.

When the Net-Net SBC completes media negotiation with the H.323 endpoint, it issues a re-INVITE to the SIP endpoint that contains the updated information needed for media transmission. In response, the SIP endpoint sends a 200 OK message that the Net-Net SBC answers with an ACK. Then RTP can flow between the two endpoints.



Status and Codec Mapping

The Net-Net SBC maps SIP and H.323 status codes as described in this section. Status and codec mapping do not require configuration; they occur transparently.

IWF Termination from H.323

When a call that requires the IWF terminates from the H.323 side, the Net-Net SBC uses the mapping scheme in the following table to determine the appropriate SIP status.

H.323 Disconnect Reason	SIP Status
No Bandwidth	480 Temporarily Unavailable
Gatekeeper Resource	404 Not Found
Unreachable Destination	404 Not Found
Destination Rejection	603 Decline
Invalid Revision	505 Version Not Supported
No Permission	401 Unauthorized

H.323 Disconnect Reason	SIP Status
Unreachable Gatekeeper	503 Service Unavailable
Gateway Resource	480 Temporarily Unavailable
Bad Format Request	400 Bad Request
Adaptive Busy	486 Busy Here
In Conference	486 Busy Here
Undefined Reason	500 Internal Server Error
Facility Call Deflection	486 Busy Here
Security Denied	401 Unauthorized
Called Party Not Registered	404 Not Found
Caller Not Registered	401 Unauthorized

IWF Termination During H.323 RAS

When a call that requires the IWF terminates from the H.323 side during RAS and generates an error, the Net-Net SBC uses the mapping scheme in the following table to determine the appropriate SIP status.

H.323 RAS Error	SIP Status
Called Party Not Registered	404 Not Found
Invalid Permission	401 Unauthorized
Request Denied	503 Service Unavailable
Undefined	500 Internal Server Error
Caller Not Registered	401 Unauthorized
Route Call To Gatekeeper	305 User Proxy
Invalid Endpoint ID	500 Internal Server Error
Resource Unavailable	503 Service Unavailable
Security Denial	401 Unauthorized
QoS Control Not Supported	501 Not Implemented
Incomplete Address	484 Address Incomplete
Route Call to SCN	302 Moved Temporarily
Aliases Inconsistent	485 Ambiguous
Not Currently Registered	401 Unauthorized

IWF Termination from SIP

When an call the requires the IWF terminates from the SIP side, the Net-Net SBC uses the mapping scheme in the following table to determine the appropriate H.323 Release Complete Reason code.

SIP Status	H.323 Release Complete Reason
300 Multiple Choices	Undefined Reason
401 Unauthorized	Security Denied
402 Payment Required	Undefined Reason
403 Forbidden	No Permission
404 Not Found	Unreachable Destination
405 Method Not Allowed	Undefined Reason
406 Not Acceptable	Undefined Reason
407 Proxy Authentication Required	Security Denied
408 Request Timeout	Adaptive Busy
409 Conflict	Undefined Reason
410 Gone	Unreachable Destination
411 Length Required	Undefined Reason
414 Request-URI Too Large	Bad Format Address
415 Unsupported Media Type	Undefined Reason
420 Bad Extension	Bad Format Address
480 Temporarily Unavailable	Adaptive Busy
481 Call/Transaction Does Not Exist	Undefined Reason
482 Loop Detected	Undefined Reason
483 Too Many Hops	Undefined Reason
484 Address Incomplete	Bad Format Address
485 Ambiguous	Undefined Reason
486 Busy Here	In Conference
487 Request Terminated	Undefined Reason
488 Not Acceptable Here	Undefined Reason
500 Internal Server Error	Undefined Reason
501 Not Implemented	Undefined Reason
502 Bad Gateway	Gateway Resource
503 Service Unavailable	Gateway Resource
504 Gateway Timeout	Adaptive Busy
505 Version Not Supported	Invalid Revision
600 Busy Everywhere	Adaptive Busy
603 Decline	Destination Rejection

SIP Status	H.323 Release Complete Reason
604 Does Not Exist Anywhere	Unreachable Destination
606 Not Acceptable	Undefined Reason

Q.850 Cause to H.323 Release Complete Reason

When an call the requires the IWF terminates from the H.323 side and no H.323 Release Complete Reason is specified, the Net-Net SBC maps the Q.850 cause to an H.323 Release Complete Reason using the mapping scheme in the following table. This new H.323 status is then mapped to a SIP status as described in the IWF Termination from SIP table.

Q.850 Cause	H.323 Release Complete Reason
No Route To Destination	Unreachable Destination
Normal Call Clearing	Destination Rejection
User Busy	In Conference
Subscriber Absent	Called Party Not Registered
Invalid Number Format	Bad Format Address
Normal Unspecified	Undefined Reason
No Circuit/Channel Available	No Bandwidth
Network Out Of Order	Unreachable Gatekeeper
Temporary Failure	Adaptive Busy
Switching Equipment Congestion	Gateway Resource
Resource Unavailable	Gatekeeper Resource
Incompatible Destination	Invalid Revision
Interworking Unspecified	No Permission

Codec Mapping

The Net-Net SBC uses the following mapping scheme when converting media specifications between H.245 (used in H.323) and SDP (used in SIP).

Media coming into the Net-Net SBC one way exits the system in the corresponding way as specified in the following table. For example, media coming into the Net-Net SBC as H.245 type g711Ulaw64k exits the system as media type PCMU.

H.245 Type	SDP Media Type
g711Ulaw64k	PCMU
g711Ulaw56k	PCMU
g711Alaw64k	PCMA
g711Alaw56k	PCMA
g726	G726-32
g7231	G723
g722	G722

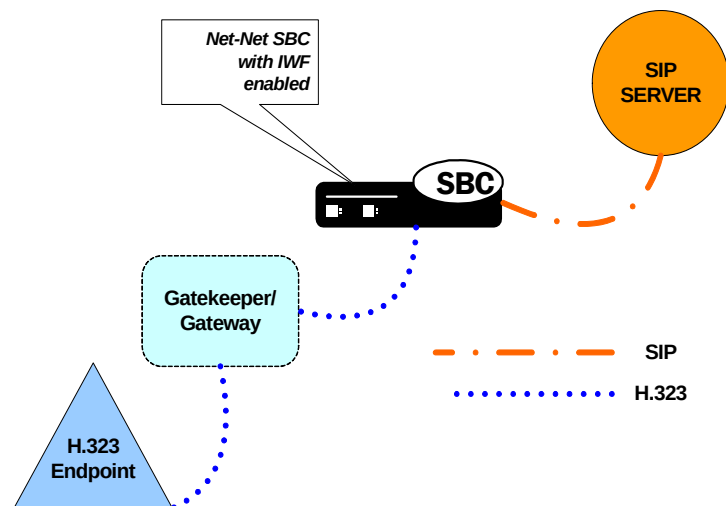
H.245 Type	SDP Media Type
g728	G728
g729wAnnexB	G729
g729	G729 fmt:18 annexb=no
h261VideoCapability	H261
h263VideoCapability	H263

IWF Service Enhancements

This section describes the Net-Net SBC features that are supported for when the Net-Net SBC performs interworking between SIP and H.323. Enabling these enhancements only requires that you set up a fully functional SIP configuration, a fully functional H.323 configuration, and that you enable IWF on your Net-Net SBC. You do not have to set any special configuration because these enhancements happen automatically.

SIP Redirect—H.323 LRQ Management

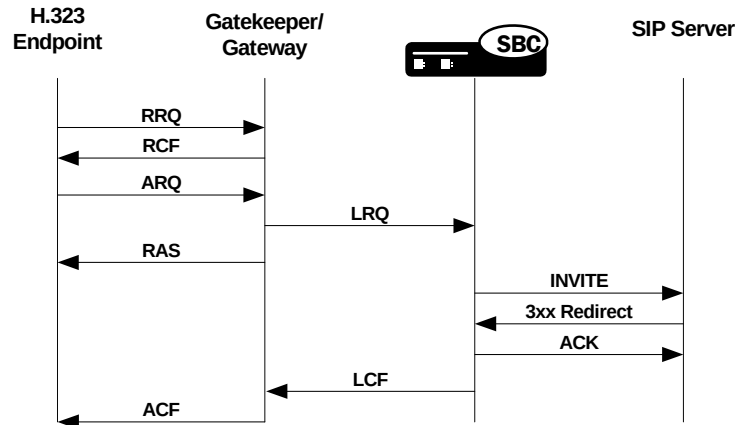
When it needs to interact with a SIP Redirect server, the Net-Net SBC can interpret the SIP messages and manage them on the H.323 side of the session. For IWF sessions, the Net-Net SBC handles SIP Redirect and H.323 LRQ messages.



Redirect—LRQ Management Sample 1

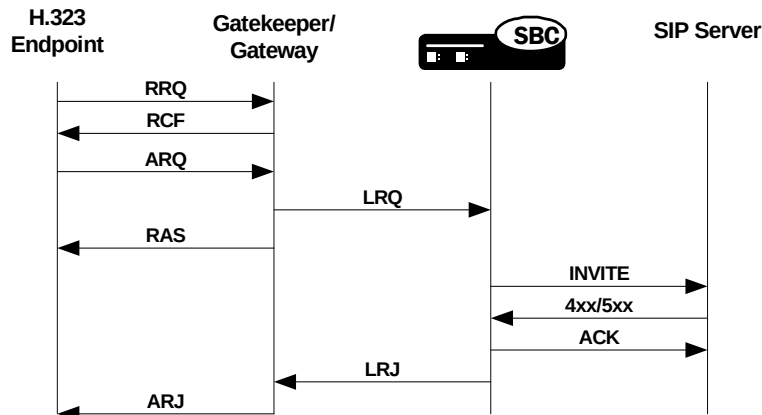
This section presents three possible scenarios for SIP Redirect-H.323 LRQ management.

The following diagram shows an established session that uses SIP Redirect—H.323 LRQ management. Here, the Net-Net SBC sends an INVITE to a SIP Redirect Server that responds with a 3xx Redirection message. The Net-Net SBC then sends the gatekeeper/gateway an LCF message that causes an ACF message to be sent to the H.323 endpoint.



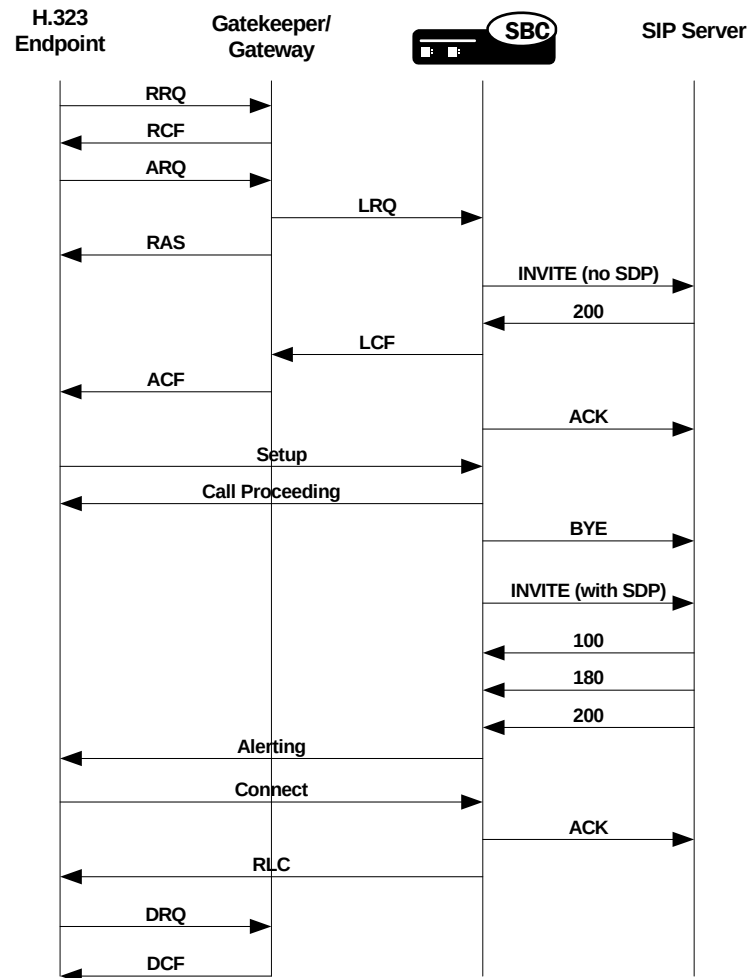
Redirect—LRQ Management Sample 2

The following diagram shows how the Net-Net SBC handles the exchange when the SIP Redirect server declares either that there is an error or that there is no such user. These SIP messages come from either the 4xx Request Failure or 5xx Server Failure series. In the example below, the SIP Redirect server returns a 401 Unauthorized message, which the Net-Net SBC interworks and communicates to the H.323 gatekeeper/gateway as an LRJ. Then the H.323 gatekeeper/gateway issues an ARJ to the H.323 endpoint.



Redirect—LRQ Management Sample 3

In this call flow, the SIP server issues a 2xx Successful message that is not supposed to be sent (because a 3xx, 4xx, or 5xx message should be sent in response to the Net-Net SBC's INVITE). The Net-Net SBC sends a BYE message to the SIP Redirect Server, but it tries to initiate the session again, this time successfully. The final sample call flow shown rarely occurs.



SIP INFO and DTMF UII Management

The Net-Net SBC supports DTMF for calls that require the IWF, enabling features such as keypress, alphanumeric, and hookflash. Because tones are not transmitted as audio, they must pass as out-of-band signaling information, meaning that the Net-Net SBC needs to convert an H.245 UII (User Input Indication) into SIP.

Depending on the capability of the H.323 endpoint, the Net-Net SBC sends either an alphanumeric or DTMF signal in the H.245 UII. The Net-Net SBC sends nothing if the endpoint does not support an alphanumeric or DTMF signal. The SIP INFO message will have a content type of `application/dtmf-relay`, and the message body will be in the form `Signal=*r\nDuration=250\r\n`. If the duration is absent in the SIP INFO or the UII received on the H.323 side is alphanumeric, the Net-Net SBC uses the a 250 millisecond default value.

Mid-Session Media Change

Mid-session media change happens during a call the requires the IWF when the type of media being sent while a session is in progress changes. For example, a fax

transmission might require mid-session media change; besides fax, other applications of this feature are possible. To support the transmission of a T.38 fax sent over an IWF session, some media channels must be opened and others closed. In addition, the Net-Net SBC can accommodate a request for media change from, for example, audio to an image type for T.38 fax.

Because the media requirements are driven by endpoints and Gateways, you do not have to configure the Net-Net SBC's mid-session media change support.

Early Media

For call that require the IWF, the Net-Net SBC supports a cut-through for early media for calls that originate in SIP or H.323.

For a session originating in SIP, the provisional message will contain the SDP information if a Fast Start OLC was received in the Call Proceeding, Alerting, or Progress messages. The same SDP will be sent in the SIP 200 OK.

For a session that starts in H.323, the Net-Net SBC translates the SDP it receives in SIP messages (either a 180 or a 183) into the appropriate H.323 Fast Start elements: Alerting or Progress. If the Alerting or Progress messages contain Fast Start elements, the Progress Indicator Q.931 information element (IE) will also be included in the message with Progress Descriptor 8, indicating that in-band information or an appropriate pattern is now available. This causes the call party to enable end-to-end early media in the reverse direction in accordance with H.323 v4.

In addition, the Net-Net SBC allows early media to flow in the forward direction for a call that requires the IWF starting in H.323 that is being translated to SIP. This happens after the Net-Net SBC has received provisional response with SDP and has sent Alerting or Progress message with Fast Start to the calling party. Similarly, early media in the forward direction is enabled for a call the requires the IWF starting in SIP and being translated to H.323. This happens after the Net-Net SBC received Alerting or Progress messages with Fast Start and maps the Alerting or Progress to SIP 180 or 183 provisional response with the SDP answer.

Display Name Mapping

The Net-Net SBC displays the full name and number of the calling party (for features such as Caller ID) when it handles calls that require the IWF. The Net-Net SBC takes the display name in the From field of the SIP INVITE and maps it to the display IE so that it can show the full name of the calling party.

IWF Ringback Support

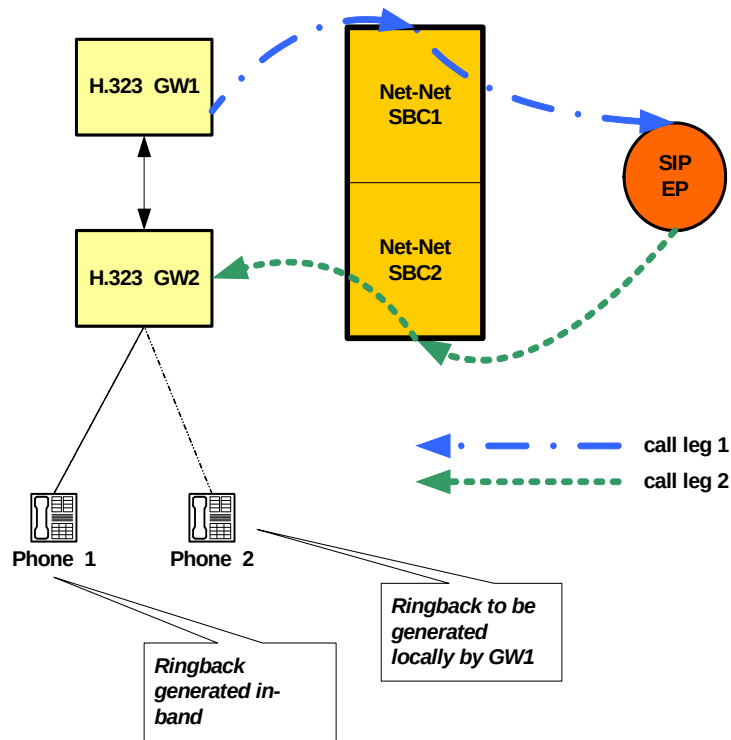
When interworking SIP and H.323 to a gateway, PSTN gateway, or other endpoint, the Net-Net SBC uses the mappings shown in the table below. The absence or presence of SDP in the SIP provisional message determines whether the tones are generated in-band or locally.

For each of the mappings listed in the following table, this section provides a sample call flow.

SIP Message	H.323 Message
No Message	CallProceeding
No Message	Progress without PI
183 with SDP	Progress with PI

SIP Message	H.323 Message
180 w/o SDP	Alert without PI
180 with SDP	Alert with PI

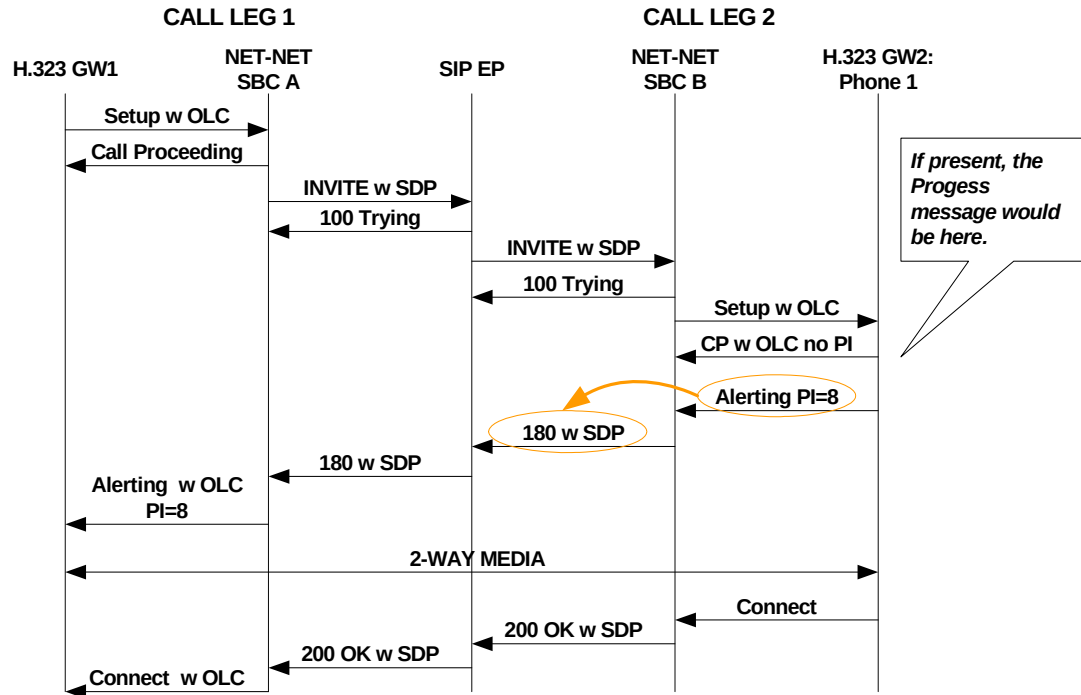
In the following diagram, a call the requires the IWF passes through the Net-Net SBC twice, creating two call legs. The call originates from H.323 GW1 and terminates in Phone 1 or Phone 2.



Sample 1: In-band Ringback without Progress Message

This sample flow shows how the Net-Net SBC handles a call that requires the IWF where there is no progress message. In this call flow, there is a progress indicator of eight (8), meaning that ringback is in-band.

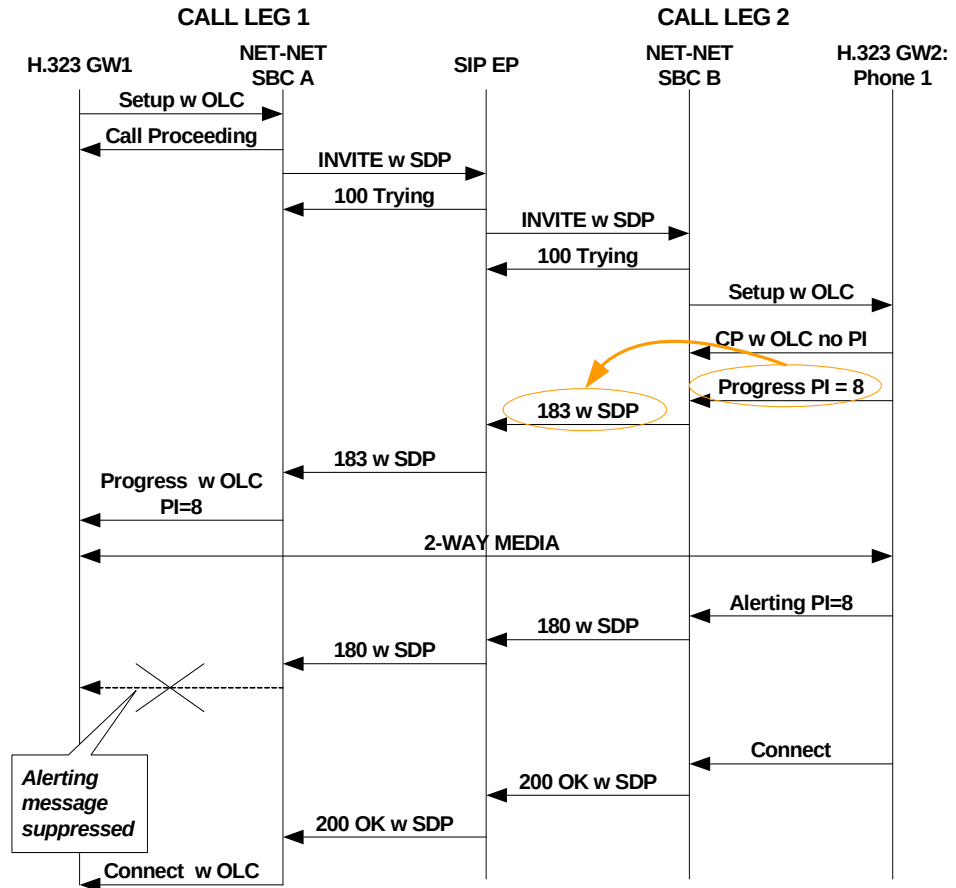
In this diagram, you can see that the Net-Net SBC maps the progress indicator included in the Alerting message sent from Phone 1 through H.323 GW2 to a SIP 180 message with SDP. When the Progress message appears, it contains the progress indicator rather than the Alerting message containing it.



Sample 2: In-band Ringback with Progress Message

This sample flow shows how the Net-Net SBC handles a call the requires the IWF where there is a progress message. In this call flow, there is a progress indicator of eight (8), meaning that ringback is in-band.

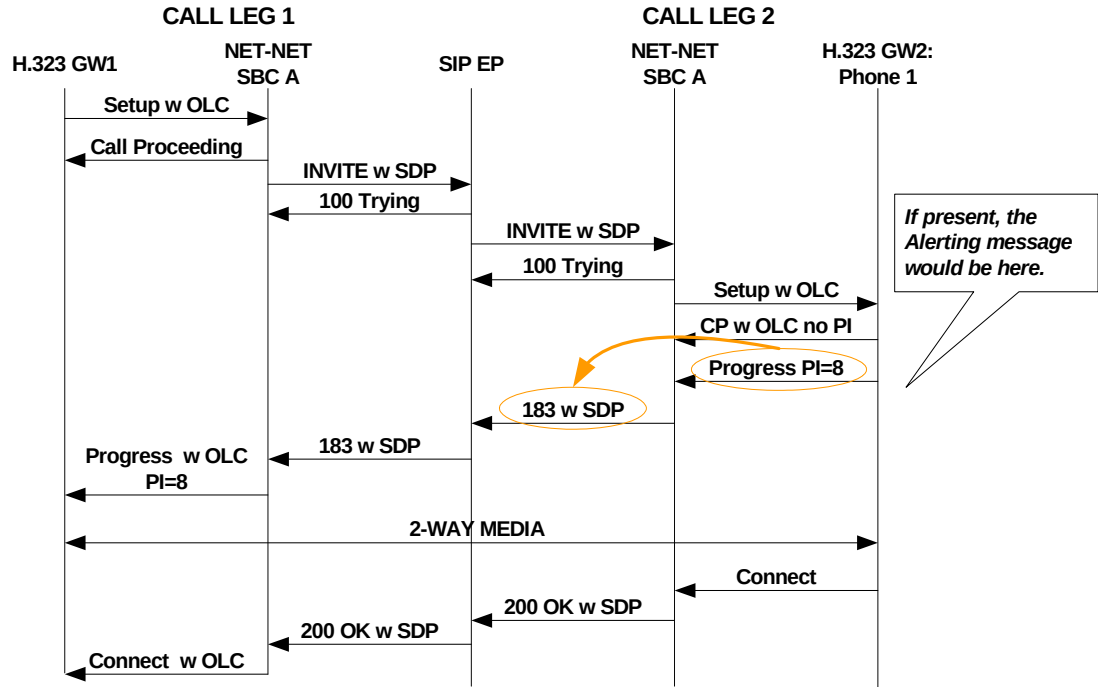
For this call flow, you can see again that the Net-Net SBC maps the progress indicator included in the alerting message sent from Phone 1 through H.323 GW2 to a SIP 180 message with SDP. Note that now the Progress message contains the progress indicator.



Sample 3: In-band Ringback without Alerting Message

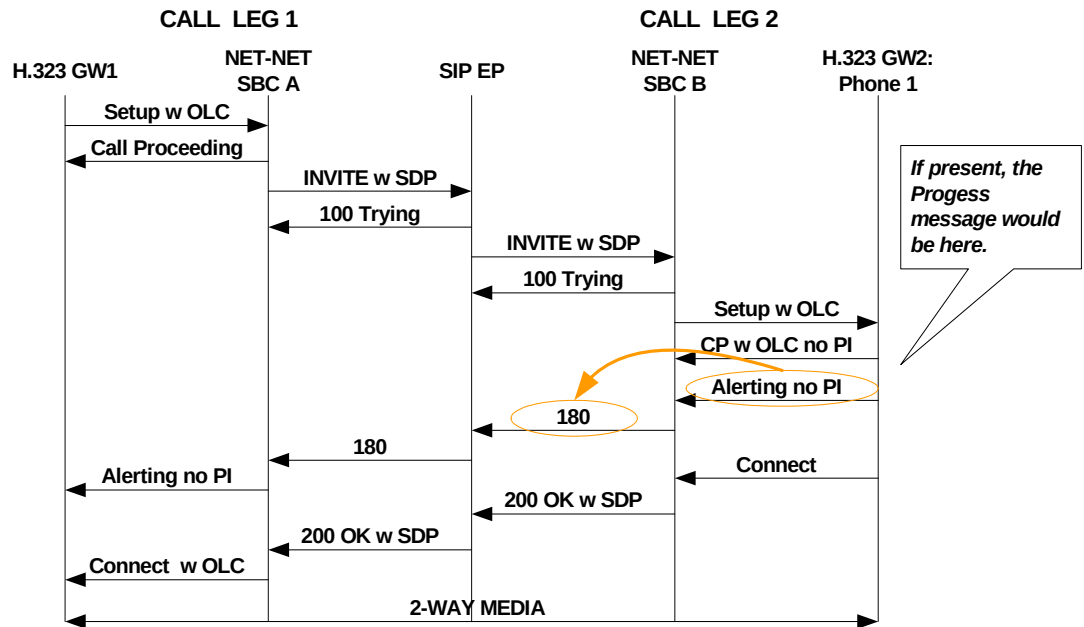
This sample flow shows how the Net-Net SBC handles a call that requires the IWF where there is no progress message. In this call flow, there is a progress indicator of eight (8), meaning that ringback is in-band.

In this diagram, you can see that the Net-Net SBC maps the progress indicator included in the Progress message sent from Phone 1 through H.323 GW2 to a SIP 180 message with SDP. When the Alerting message appears, it contains the progress indicator rather than the Progress message containing it.



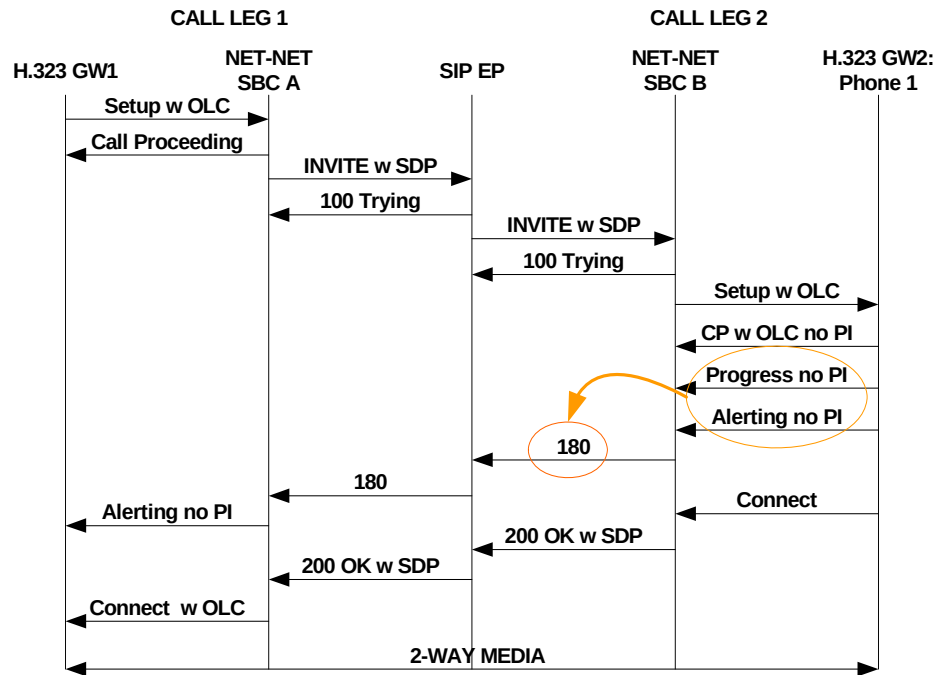
Sample 4: Out-of-band Ringback without Progress Message

When there is no progress indicator included in the Alerting message, then there is out-of-band ringback. The Net-Net maps the Alerting message to a SIP 180, but it does not include SDP in the SIP 180. This call flow shows that there is no Progress message and that media cannot be set up until after H.323 Connect and SIP messages are sent.



Sample Flow 5: Out-of-band Ringback with Progress Message

When there is no progress indicator included in either the Alerting or Progress messages, then there is out-of-band ringback. The Net-Net maps the Alerting message to a SIP 180, but it does not include SDP in the SIP 180. This call flow shows includes the Progress message; still, media cannot be set up until after H.323 Connect and SIP messages are sent.

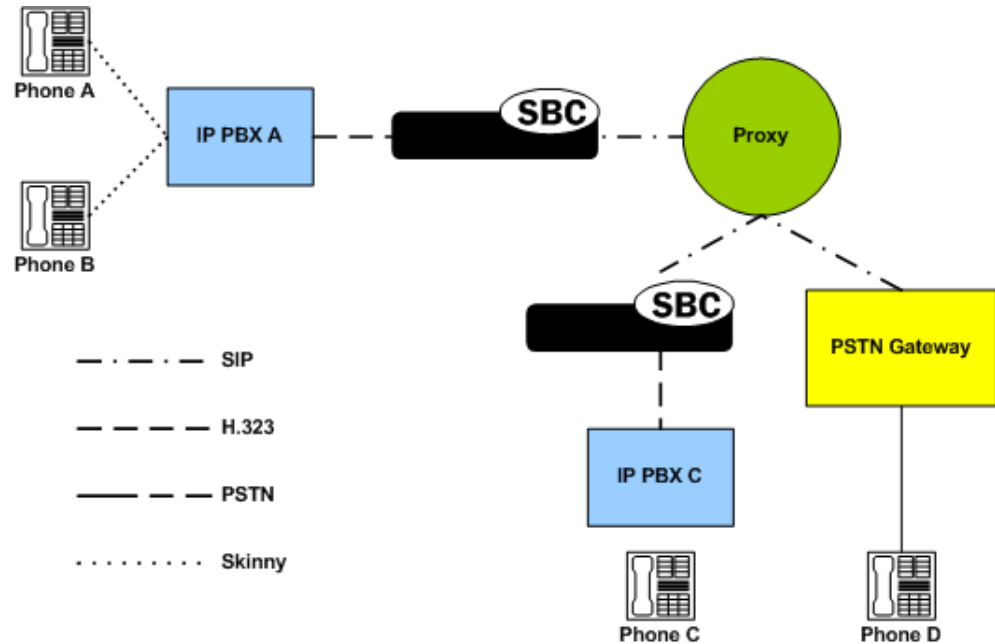


H.323 Endpoint- Originated Call Hold and Transfer

When calls that require the IWF originating in H.323, the Net-Net SBC supports call hold, transfer, and conference for the H.323 call leg. The call hold and transfer feature uses signaling procedures based on the ITU-T recommendations/H.323 specification for “third party initiated pause and rerouting.”

You do not have to configure the Net-Net SBC’s call hold and transfer feature.

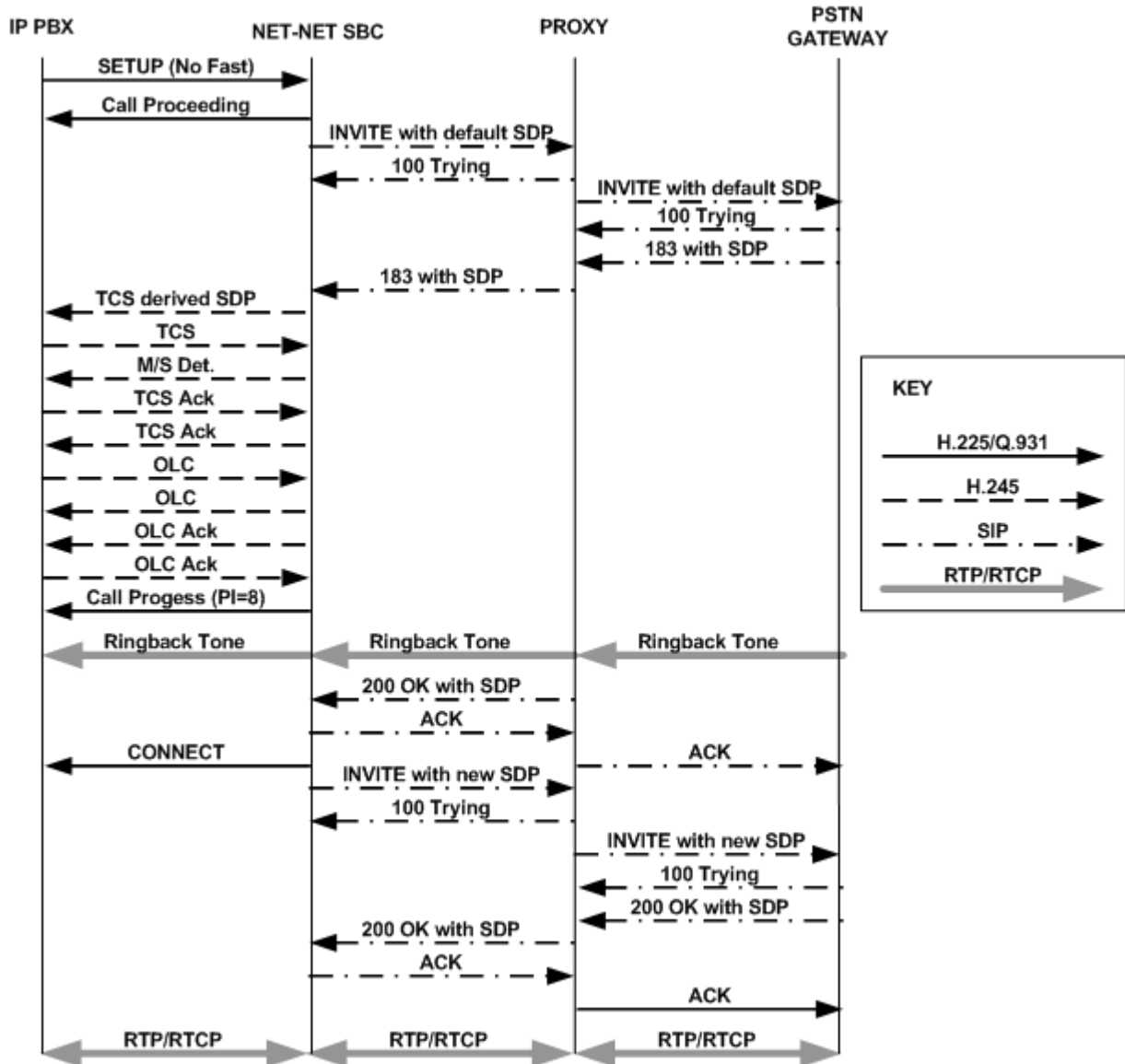
The following diagram shows how the Net-Net SBC provides call hold and transfer support for IWF sessions that originate in H.323. As you review this section’s call flow diagrams, you might want to refer back to the following logical diagram directly below to review the network elements involved, and what protocols they use.



Basic Call

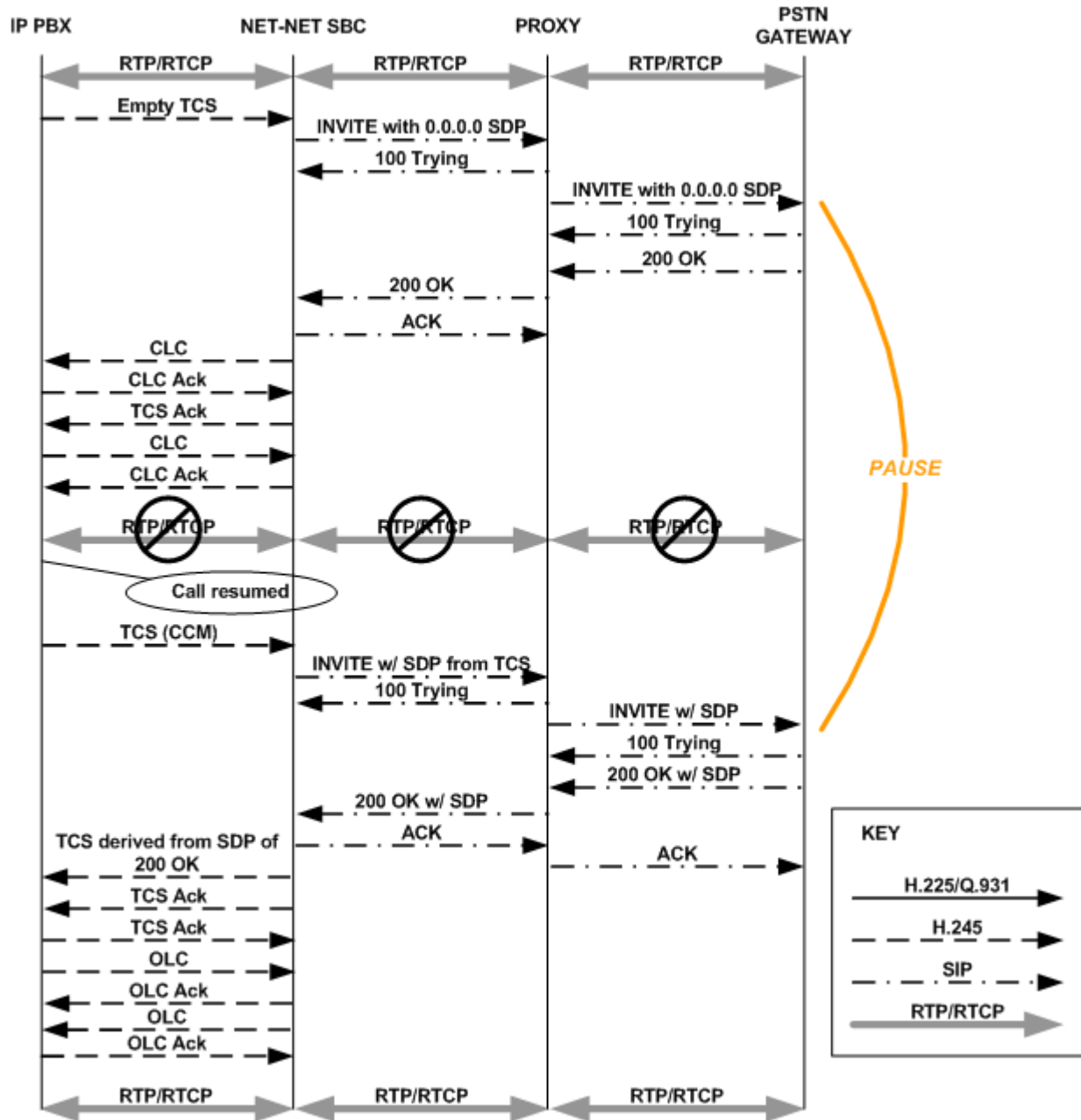
In the following sample basic call, IP PBX A sends an H.323 Slow Starts message ultimately destined for the PSTN through the Net-Net SBC. The Net-Net SBC performs translation to SIP and inserts default information about media. Once the PSTN gateway responds with a 183 containing SDP, the Net-Net SBC sends that information to IP PBX A. Then the Net-Net SBC and the IP PBX exchange TCS- and OLC-related messages, and they negotiate master-slave determination. The Net-Net SBC also sends IP PBX A a Call Progress message with a progress indicator of 8.

After the ringback tone, the proxy sends a 200 OK message with SDP to the Net-Net SBC. The Net-Net SBC sends a Connect message to the IP PBX A, and then it sends another SIP INVITE to the proxy that contains amended SDP (if that information about media is different from the default). After 200 OK and ACK messages are exchanged, media (RTP/RTCP) flow takes place.

**Hold**

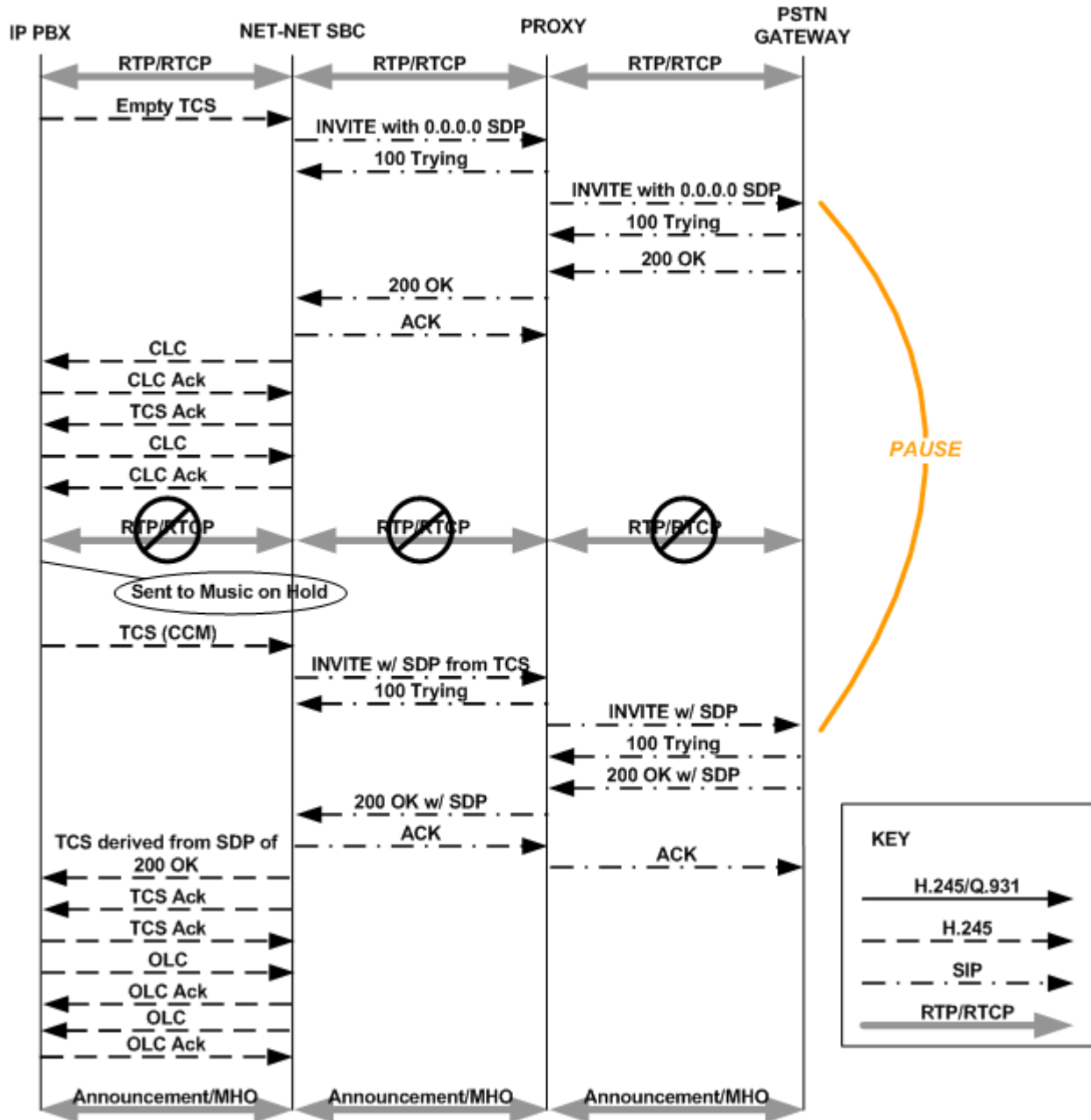
This sample call flow assumes that the IWF call is established and that the RTP/RTCP flow is already in progress. The hold button is pushed, and IP PBX A sends an empty TCS to the Net-Net SBC. The Net-Net SBC puts the called party on hold by sending an INVITE message with 0.0.0.0 SDP to the SIP side of the call. Using 0.0.0.0 as the media address effectively stops the media flow. This INVITE is acknowledged, and the Net-Net SBC closes the channels on the H.323 side, halting the RTP/RTCP flow.

When the caller on the H.323 side takes the call off hold, it resumes with a TCS that the Net-Net SBC receives and then translates on the SIP side as an INVITE with SDP. After that INVITE is acknowledged and received, the Net-Net SBC opens logical channels on the H.323 side and RTP/RTCP flows resume.



Music On Hold

This scenario is similar to the hold feature enabled for calls that require the IWF, except that after the RTP/RTCP flow between the H.323 and SIP sides stops, the call is sent to music on hold. Before the announcement or music plays, the Net-Net SBC sets up the necessary support for media to be exchanged.



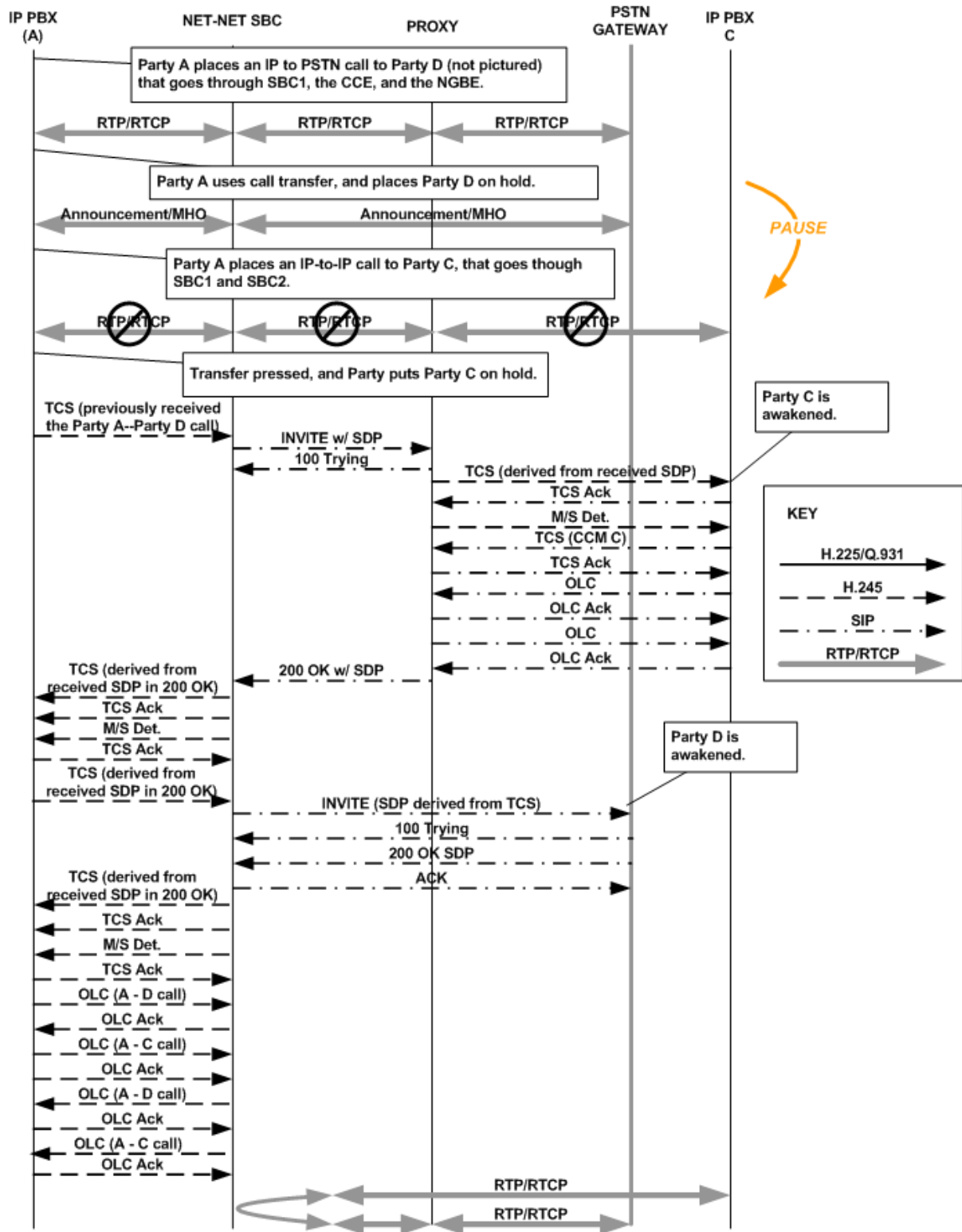
Transfer

The call flow described in this section recalls the diagram at the top of the H.323 Endpoint-Originated Call Hold and Transfer section, where endpoints A, B, and C are H.323 devices and endpoint D is a SIP device. When you follow the signaling and media flows, note that there are two Net-Net SBCs in the call transfer and two sets of SIP/H.323 translations that take place. The first Net-Net SBC translates H.323 to SIP, and the second performs the same operations with the protocols reversed.

In the scenario pictured, Party A is on a call with Party D, but wants to transfer Party C to Party D. Party A places Party D on hold, and then makes the call to Party C. Party A then puts Party C on hold, pressing the transfer button. You can see that Net-Net SBC1 receives a TCS from the IP PBX, which is then translated to SIP. Net-Net SBC2 receives it, performs the required protocol translations, and then opens a session with Party C via another IP PBX. Once this session is up and Party D is awakened, channels are established for media exchange.

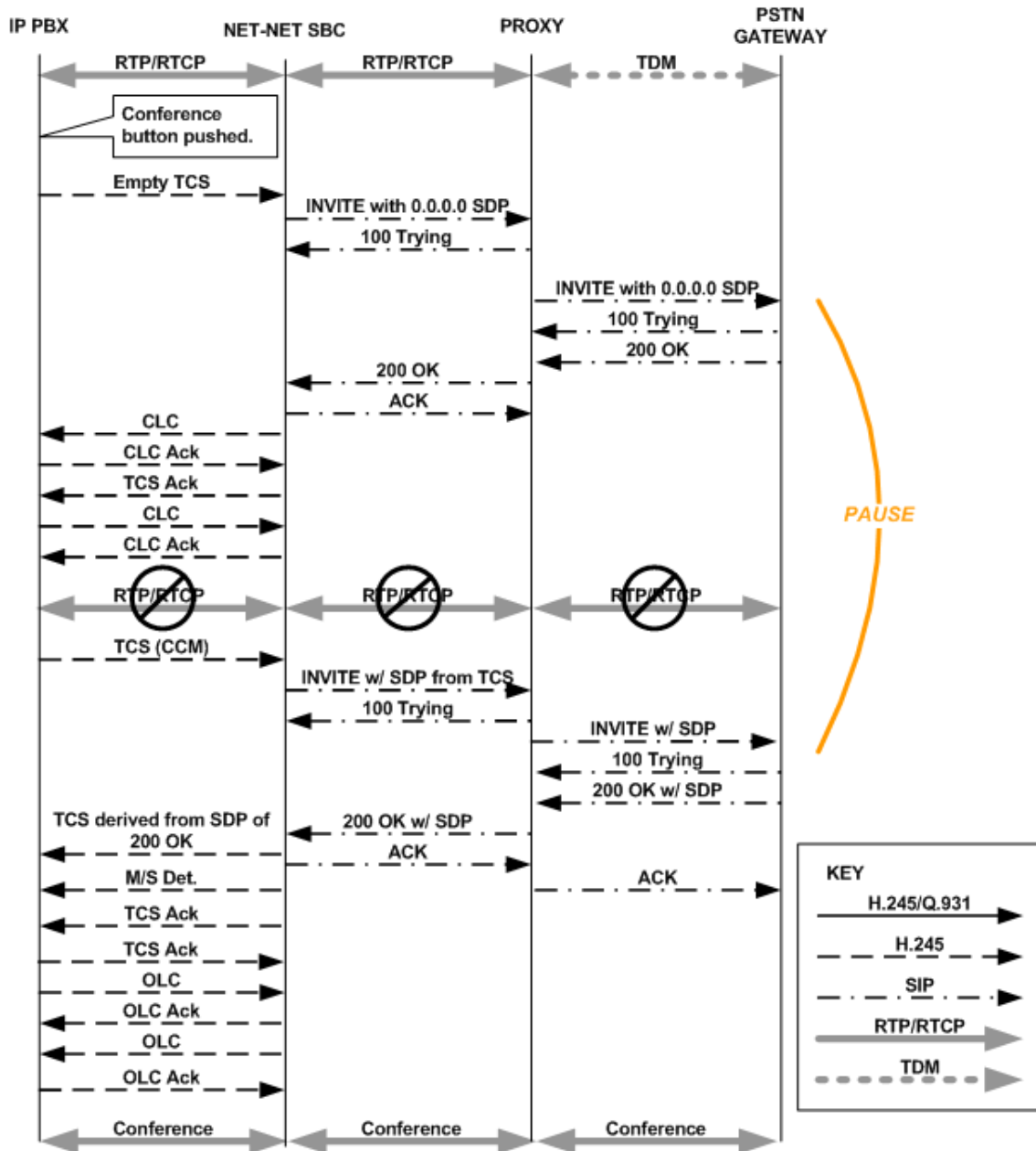
In order to redirect the media so that it flows between Party C and Party D, the Net-Net SBC1 and IP PBX C exchange OLC and OLC Ack messages that contain address information for Party C and for Party D. Address information for both parties is contained in the OLC Ack messages that the Net-Net SBC exchanges with the IP PBX. IP PBX A does not move forward with the call until it has the necessary address information.

Even though Party A's participation in the call stops early in this scenario, the IP PBX with which it is associated keeps the signaling sessions with the Net-Net SBC alive to manage the transfer.



Conference

To conference a call the requires the IWF that starts in H.323, the Net-Net SBC uses a scenario much like the one used for holding a call the requires the IWF. Here again, the INVITE with 0.0.0.0 as the media address and the closing of logical channels stops the flow of RTP/RTCP. After signaling and SDP/media information are re-established, RTP/RTCP for the conference flows.



IWF Call Forwarding

This section describes the Net-Net SBC's IWF Call Forwarding feature, which is supported for calls initiated in SIP that require interworking to H.323.

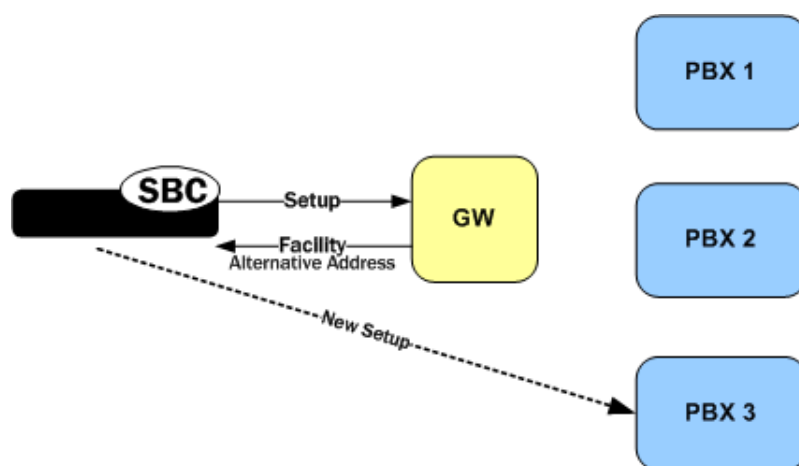
Prior to the implementation of this feature, the Net-Net SBC did not forward calls when the remote H.323 endpoint sent a Facility message with Call deflection as the reason and an alternate address for forwarding. Instead, it would either:

- Fail to release the initial call and initiate the forwarded call
- Drop the entire call when the remote endpoint for the call tore down the session

New Behavior

In the diagram below, you can see that the Net-Net SBC sends the initial Setup message to the gateway, and the gateway returns the Facility message with an alternate address for forwarding. Rather than engaging in its former behavior, the Net-Net SBC now releases the call with the gateway and sends a new Setup to the alternate address from the Facility message.

This new Setup up has no effect on the first call leg, which remains connected.



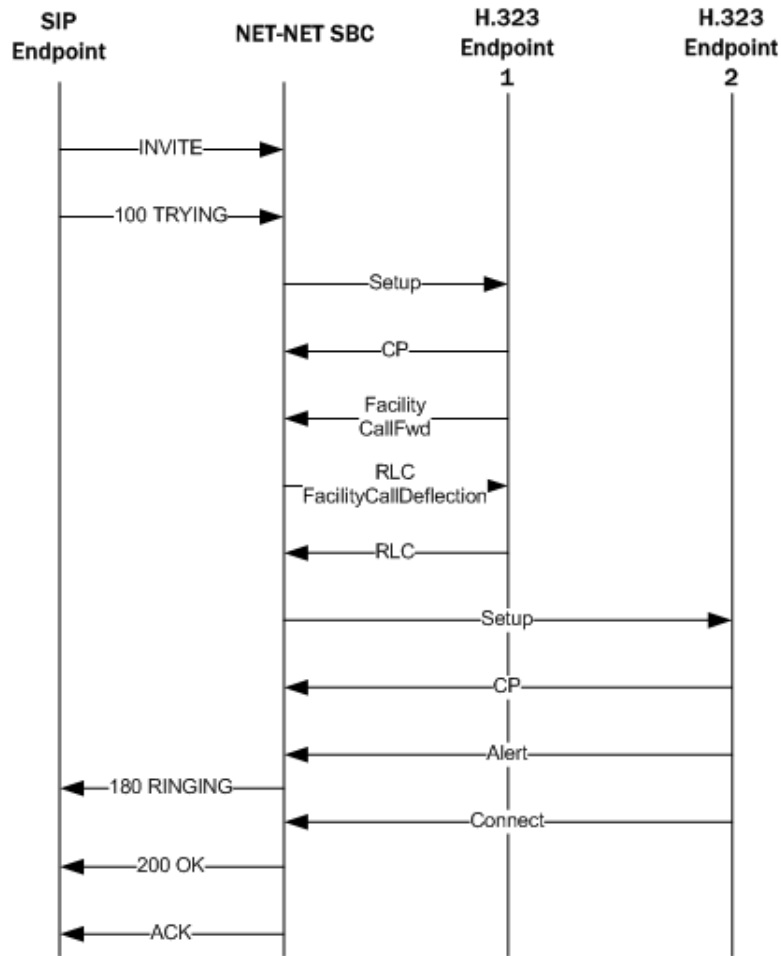
How It Works

When it receives a Facility message with the reason CallForwarded, the Net-Net SBC looks for an alternate transport address in the Facility's alternativeAddress or alternativeAliasAddress element. The Net-Net SBC releases the egress call with the reason facilityCallDeflection. Then it takes one of two courses of action:

- If it does not find an alternative address, the Net-Net SBC releases the ingress call (with 486 BUSY HERE for a call being interworked from SIP to H.323).
- If it finds an alternative address and the egress call has not been alerted or answered, the Net-Net SBC at this point tries to initiate a new egress call. The Net-Net SBC uses the alternative alias address to populate the calledPartyNumber information element (IE) and the destination address of the new Setup.

H.323 Sample Call Flow

The following diagram shows how the H.323 Call Forwarding feature works in a purely H.323 environment.



Media Release for H.323 SS-FS Calls for IWF

When the Net-Net SBC routes a slow-start to fast-start call, it is possible for the same fast-start call to be routed back through the Net-Net SBC making for a hairpin flow. If it does become a hairpin flow, then the Net-Net SBC routes it to its destination as a fast-start to fast-start call. This can result in one-way media if:

- The destination of the hairpin call is in the same realm as the originating slow-start to fast-start call
- The realm reference in the first bullet item is configured to disable in-realm media management
- The called endpoint accepts the proposed fast-start logical channels

The enhancements to the Net-Net SBC's behavior described in this section show how the Net-Net SBC follows additional procedures when setting up a hairpin flow to avoid one-way media when media release occurs.

How It Works for H.323

For H.323 calls, the Net-Net SBC establishes media using the H.245 procedures described in the H.245 ITU-T recommendation: control protocol for multimedia communication. It also uses the Fast Connect procedure defined in the H.323 ITU-T recommendation: packet-based multimedia communication systems.

The latter ITU-T recommendation allows a calling endpoint to send a Setup message that contains a fastStart element, a sequence of OLC structures that describe the calling endpoint's proposed forward/reverse logical channels. If the called endpoint accepts this proposal, then logical channels are established.

When the Net-Net SBC translates a call originating in slow-start to fast-start, it uses a Fast Connect procedure in the outgoing leg by sending an outgoing Setup that includes a fastStart element with one or more OLC structures. But when the Net-Net SBC constructs this message, it is unaware of whether the call will become hairpinned or if media release will occur. Because it does not yet have this information, the Net-Net SBC sets the Network Address and the TSAP identifier in the OLC structures to the ingress IP address and port of a corresponding media flow allocated for media traveling between the calling and called endpoints. So if the called endpoint accepts the fastStart the Net-Net SBC proposes, the called endpoint would send its media to the Net-Net SBC. After acceptance, the Net-Net starts H.245 procedures on the slow-start side of the call to set up logical channels on that side. Then the Net-Net SBC updates the IP address and port of the media flows using OLC and OLCAck messages received from the calling endpoint.

This procedure works well for endpoints that are not in the same realm, or that are in the same realm for which media management is disabled, because each endpoint must send its media through the Net-Net SBC. When the endpoints are in the same realm and when media management is enabled, however, the Net-Net SBC must perform additional steps for media release in slow-start to fast-start calls.

To support media release in slow-start to fast-start calls, the Net-Net SBC performs a hold-and-resume procedure on the fast-start side. After it establishes channels on the slow-start side and if it detects media release being enabled, the Net-Net SBC sends an empty TCS to the fast-start side to put that side on hold. Then the called endpoint closes all the logical channels it previously opened in the Fast Connect procedure and stops transmitting to them. And the Net-Net SBC also closes its logical channels. Once the channels are closed, the Net-Net SBC resumes the call by sending a new, restricted TCS to the fast-start side. The restricted TCS only contains the receive and transmit capabilities of the codec types that the called endpoint accepted in the Fast Connect procedure, and it forces the called endpoint to re-open logical channels of the same codec types accepted in the Fast Connect procedure. Once it receives an OLC from the called endpoint, the Net-Net SBC sends an OLCAck with the Network Address and TSAP identifier for the logical channel from the calling endpoint. Then the Net-Net SBC re-opens logical channels (of the same codec types that it opened in the Fast Connect procedure). If the called endpoint has not changed its Network Address and TSAP identifier for its logical channels, media is re-established after the Net-Net SBC and the called endpoint exit the hold state. The last step is for the Net-Net SBC to re-send the full TCS message from the calling to the called endpoint to inform the called endpoint of the full capabilities of the calling endpoint.

Hold-and-Resume Procedure

The hold-and-resume procedure has three states:

- **Media Hold**—Starts when the Net-Net SBC sends the empty TCS to the called endpoint to put it on hold.

When it detects media release, the Net-Net SBC puts the called endpoint on hold. It can only do so if it has exchanged the TCS/TCSAck messages and completed master-slave determination with the calling endpoint.

When the Net-Net SBC receives a TCSAck in response to the empty TCS that it sent to the called endpoint, it closes the logical channels it opened as part of the Fast Connect procedure; the called endpoint likewise closes its logical channels. The two then exchange CLC and CLCAck messages, which signals the start of the Media Resume state.

- **Media Resume**—Starts when the Net-Net SBC sends a restricted TCS to resume the call.

The restricted TCS the Net-Net SBC sends contains only the receive/transmit capabilities of the codec types previously accepted by the called endpoint in the Fast Connect procedure. This forces the called endpoint to re-open logical channels of the same codec type that were previously accepted in the Fast Connect procedure.

After sending this TCS, the Net-Net is ready (as specified in the ITU-T recommendations) to take part on the master-slave determination (MSD) process. However, the called party and not the Net-Net SBC initiates the MSD if it is required. The MSD is completed if necessary. Alternately, the called endpoint can start to re-open its logical channels. When it receives the first OLC from the called endpoint, the Net-Net SBC also starts to re-open its logical channels.

- **Media Complete**—Starts when all the logical channels that the Net-Net SBC re-opens are acknowledged by the called endpoint.

When it enters the Media Complete state, the Net-Net SBC updates the called endpoint with the full capabilities of the calling endpoint by sending the full TCS.

Additional IWF Steps

For calls originating in slow-start H.323 that require interworking to SIP, the Net-Net SBC also takes additional steps for media release in hairpinned flows that the Net-Net SBC routes as SIP to fast-start H.323.

For such a call, after the Net-Net SBC has established logical channels on the slow-start H.323 side of the call, it sends a reINVITE on the SIP side. This reINVITE has an updated session description to correct the media connection information. The Net-Net SBC performs the hold-and-resume procedure on the fast-start side of the call. This procedure re-establishes the logical channels between the Net-Net SBC and the called endpoint, avoiding the one-way media problem.

When you are configuring H.323 globally on your Net-Net SBC, you might choose to set the noReInvite option. This option stops the Net-Net SBC from sending a reINVITE after the logical channels are established on the slow-start H.323 side of the call. Instead, the Net-Net SBC's H.323 task communicates internally with its own SIP task a SIP UPDATE message that corrects the SDP; then the SIP task updates media flow destinations. But the Net-Net SBC does not send the UPDATE to the next hop, which can result in the one-way media problem if the call is hairpinned and media release occurs. For such cases, the default behavior for the noReInvite option is overridden. When the Net-Net SBC detects media release in an

H.323-SIP call, it forwards the UPDATE to the next hop even when you enable the noReInvite option.

Dependencies

This feature depends on:

- The H.323 endpoint supports the third-party-initiated pause and re-routing feature.
- The H.323 endpoint does not change its Network Address and TSAP identifier when it re-opens the logical channels.
- The H.323 endpoint does not immediately tear down the call when there is not established logical channel in the call.
- The fact that the SIP endpoint supports the UPDATE message if the noReInvite option is enabled.

Before You Configure

The Net-Net SBC's IWF requires that there be complete configurations for both SIP and for H.323. These two sets of configurations function together when the interworking is configured and enabled.

You enable the Net-Net SBC's interworking capability when you set the IWF configuration's state parameter to **enabled**, and all required H.323 and SIP configurations are established. This means that all of the following configurations must be established:

- A full SIP configuration, including SIP interfaces, SIP ports, SIP-NATs (if needed), and SIP features
- A full H.323 configuration, including H.323 global and H.323 interface configurations
- Local policy and local policy attributes (the IWF will not work without these configurations)
- Media profiles
- Session agents and, if needed, session agent groups

H.323 Configuration

You must have a complete configuration to support H.323 traffic on your Net-Net system, including any required support for H.323 Fast Start or Slow Start. In the H.323 interface configuration, you are able to configure interfaces that enable communication between H.323 devices (for audio, video, and/or data conferencing sessions).

If you know that your Net-Net SBC will be handling traffic originating in Slow Start H.323, you must establish the appropriate list of media profiles in the IWF configuration. Handling Slow Start traffic also requires that you establish appropriate local policy (and local policy attribute) configurations, but configuring session agents and session agent groups is optional.

SIP Configuration

SIP functionality must also be configured on your Net-Net SBC that will perform IWF translations. You must use appropriate local policy (and local policy attribute) configurations, but configuring session agents and session agent groups is optional.

If you use session agents, then you must also configure the information you need for media profiles.

For further information about configuring the SIP signaling on your Net-Net SBC, refer to this guide's SIP Signaling Services chapter.

The Role of Route (Local) Policy

You must configure route (local) policies (and route policy attributes, if necessary) in order for translations between SIP and H.323 to take place. These route policies determine what protocol is used on the egress side of a session. Route policy and route policy attribute configurations make routing decisions for the session that are based on the next hop parameter that you set. The next hop can be any of the following:

- IP address of a specific endpoint
- Hostname or IP address of a session agent
- Name of a session agent group

You can use the application protocol parameter in the route policy attributes configuration as a way to signal the Net-Net SBC to interwork the protocol of an ingress message into a different protocol as it makes its way to its egress destination (or next hop).

For example, if you set the application protocol parameter to SIP, then an inbound H.323 message will be interworked to SIP as it is sent to the next hop. An inbound SIP message would travel to the next hop unaffected. Likewise, if you set the application protocol parameter to H.323, then an incoming SIP message will be interworked to H.323 before the Net-Net SBC forwards it to the next hop destination.

Route Policy in an IWF Session Initiated with H.323

In a session where the Net-Net SBC is interworking H.323 to SIP, it internally forwards the session on for interworking when:

- The next hop in the route policy is configured as a SIP session agent
- The next hop in the route policy is configured as a SIP session agent group
- The next hop in the route policy is not configured as a session agent or session agent group, and the application protocol parameter is set to SIP in the route policy attributes configuration.

Route Policy in an IWF Session Initiated with SIP

In a session where the Net-Net SBC is interworking SIP to H.323, it internally forwards the session on for interworking when:

- The next hop in the route policy is configured as an H.323 session agent
- The next hop in the route policy is configured as an H.323 session agent group
- The next hop in the route policy is not configured as a session agent or session agent group, and the application protocol parameter is set to H.323 in the route policy attributes configuration

In this case the route policy should also define the egress realm, which you can set in the realm parameter of the local policy attributes configuration.

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by	admin@console
Last modified date	2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2_UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Configuring Interworking

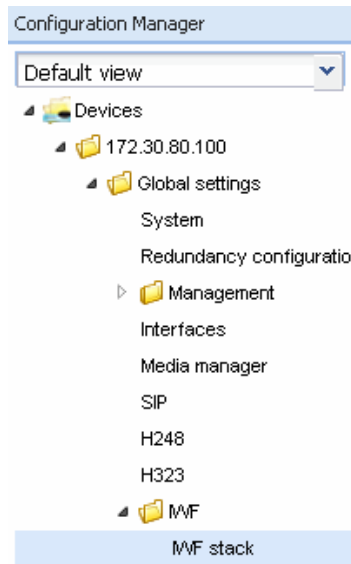
If you have already completed the steps outlined in this chapter's IWF Service Enhancements section, then enabling the IWF is a simple process. This section shows you how to enable the IWF, and how to enable certain features that you can use to supplement basic IWF functionality.

Configuring Interworking

This section describes how to enable IWF on your Net-Net SBC from the Net-Net Central.

To configure interworking:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click IWF stack. The IWF data window appears in the content area.
3. **State**—Select the IWF state from the combination box.

State

The dropdown menu for 'State' is open, showing 'enabled' as the selected option. Other visible options are 'enabled' and 'disabled'.

4. **Process log state**—Select the IWF logging state from the combination box.

Process log state

The dropdown menu for 'Process log state' is open, showing 'enabled' as the selected option. Other visible options are 'enabled' and 'disabled'.

5. **Media profiles for interworking incoming H323 slow start call to outgoing SIP call**—Click Add to add media profiles.

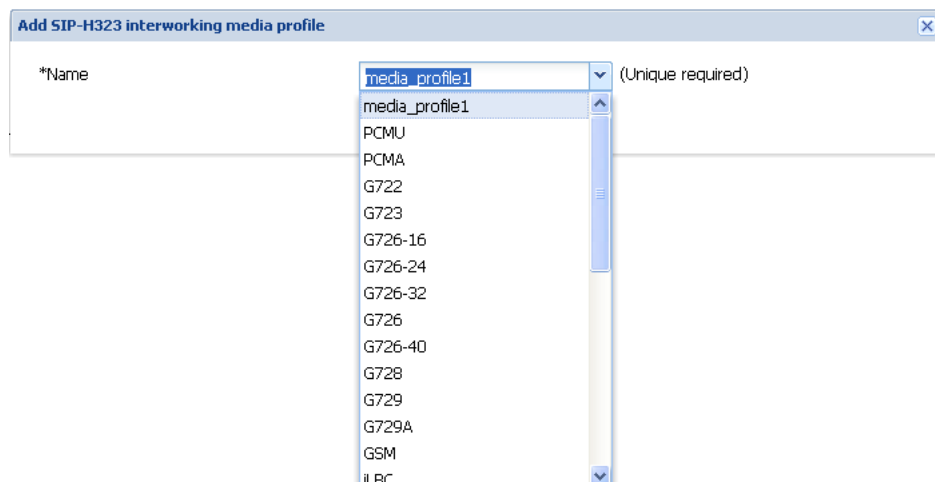
SIP-H323 interworking media profile

The window titled 'Add SIP-H323 interworking media profile' is shown. It has a large empty text area for input and two buttons at the bottom: 'Add' and 'Delete'.

The Add SIP-H.323 Interworking media profile window appears.

6. **Media profile name**—Select the media profile you want to add to the IWF service from the combination box. This name is either the name of an SDP codec (such as PCMU), or it can be *telephone-event* if you are configuring your system for DTMF support.

If you want to use more than one media profile for SIP/H.323 translations, add additional media profiles as shown in Step 5.



7. Click **OK** to continue.
8. If you want to configure RFC 3326 support (mappings for SIP status to Q.850 cause code mappings, refer to this chapter's RFC 3326 Support section for details. Otherwise, click **Apply** at the bottom of the SIP-H.323 IWF tab to complete configuring IWF.

DTMF Support

For calls that require the IWF, you can enable support for the relay of RFC 2833 DTMF digits. The availability of this feature means that the Net-Net SBC is compliant with RFC 2833, which defines two payload formats for DTMF digits. To learn more about this RFC, refer to <http://www.ietf.org/rfc/rfc2833.txt>.

Until the exchange of TCS messages with the H.323 endpoint, the Net-Net SBC has no information about the endpoint's RFC 2833 capabilities. The Net-Net SBC adds *telephone-event* to the SDP on the SIP side of the call.

For calls that require SIP/H.323 translation, you can enable support for the relay of RFC 2833 DTMF digits.

To use this feature, you need to configure a media profile called *telephone-event* and set relevant parameters for it. Application of the media profile can happen either in a session agent configuration or in the IWF configuration.

- The **name** parameter in the media profiles configuration
- The **media-profiles** list in the IWF configuration
- The **media-profiles** list in the session agent configuration

All of the scenarios outlined here assume that you have established a *telephone-event* media profile configuration.

You can configure DTMF support using the following parameters. The way that the Net-Net SBC uses these values is described below. The payload type, part of the media profiles configuration, is dynamic and varies with different endpoints, so there is no default configuration for the *telephone-event* media profile.

The *telephone-event* media profile is used as follows in these types of IWF sessions:

- **Calls that require the IWF originating in H.323 Fast Start**—The Net-Net SBC uses the channels defined in the Fast Start messages to generate SDP on the SIP side of the session.
 - If the incoming H.323 endpoint is an H.323 session agent and the media profiles parameter for the session agent is set to *telephone-event*, the Net-Net SBC will use the information in that media profile to add the *telephone-event* in the SDP.
 - If the incoming H.323 endpoint is not a session agent, the media profile set in the IWF configuration will be used.
- **Calls that require the IWF originating in H.323 Slow Start**—There is no channel (media) information available on the H.323 side.
 - If the incoming H.323 endpoint is configured as a session agent on the Net-Net SBC, then the *telephone-event* parameter in the media profiles set for that session agent configuration will be used in the SDP on the SIP side of the session.
 - If the H.323 endpoint is not a session agent or the *telephone-event* media profile is not configured in the session agent configuration corresponding to the endpoint, then the Net-Net SBC refers to the media profile information configured for the IWF configuration.
- **Calls that require the IWF originating in SIP**—If the TCS was not exchanged before a 200 OK was sent on the SIP side, the Net-Net SBC will behave in one of these two ways.
 - If the outbound H.323 endpoint is configured as a session agent, then the media profiles from that session agent configuration will be used.
 - If the outbound H.323 endpoint is not configured as a session agent, the media profile configured within the IWF configuration with the *telephone-event* value will be used.

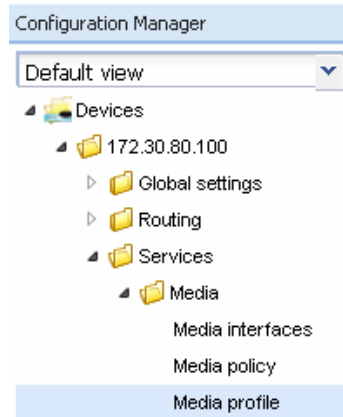
As mentioned above, DTMF support is configured by using a combination of the *telephone-event* media profile and either the session agent or IWF configuration. First you set up the media profile, then you apply it to a session agent or to the IWF configuration.

Configuring DTMF Support

DTMF support requires you to configure a media profile named *telephone-event*. This section shows you how to set it up.

To configure a *telephone-event* media profile:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**. The Add media profile dialog box appears.
4. **Media profile name**—Enter a telephone-event as the name for the profile.
5. **Subname**—Enter a subname to distinguish this media profile.

The 'Add Media profile' dialog box is shown. It has a title bar with a close button. Inside, there are two fields:

- *Name: A dropdown menu with 'telephone-event' selected. To the right of the dropdown is the text '(Unique required)'.
- *Subname: A text input field containing 'test'. To the right of the field is the text '(Unique required)'.

At the bottom of the dialog are two buttons: 'Apply' and 'Cancel'.

6. Click **Apply**.
The telephone-event profile appears under the Media profile category.
7. Select the telephone-event media profile and click **Edit**.
8. **Media type**—Retain the default Media type set of **audio**.
9. **Payload type**—Set the Payload type to **101**, which is the dynamic payload type needed to support this feature.
10. **Transport protocol**—Retain the default Transport protocol of **RTP/AVP**.
11. **Frames per RTP packet**—Retain the default Frames per RTP packet value of **0**.
12. **Bandwidth requirements (Kbps)**—Set the Bandwidth requirements to **0**.

13. **Parameters**—Enter the Parameters to be applied for the codec; these are the digits that endpoints can support.

*Name	telephone-event
*Subname	test
Media type	audio
Payload type	101
Transport protocol	RTP/AVP
Frames per RTP packet	0
Bandwidth requirements (Kbps)	0
Parameters	
	Add Delete

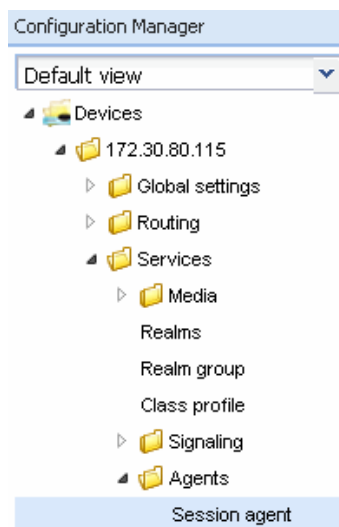
14. Click **Apply**.

Applying the Media Profile

After you have configured the *telephone-event* media profile, you need to apply it either to a H.323 session agent or the global IWF configuration.

To use DMTF support on a per-session-agent basis:

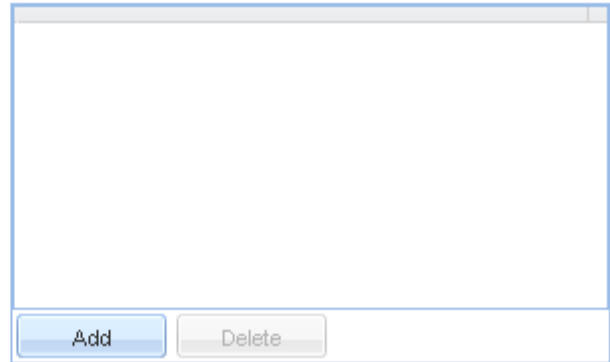
1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.

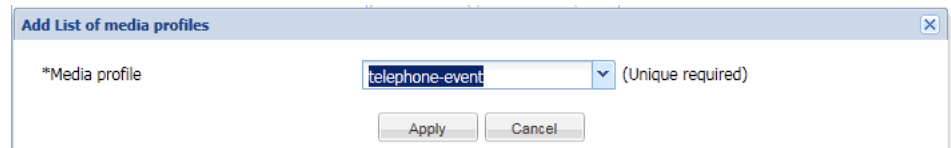
3. Click the H.323 session agent for which you want to support DTMF and click **Edit**.
4. Click **Add** in the Media profiles.

List of media profiles



The Add SA H.323 media profile dialog box appears.

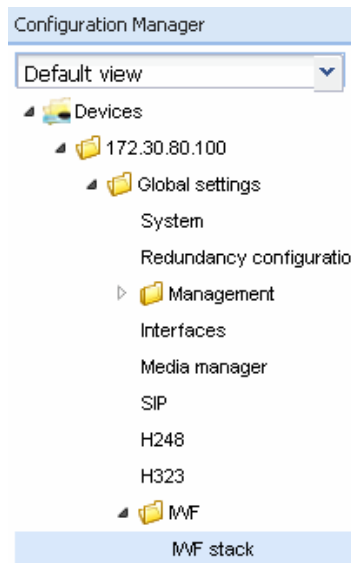
5. **Media Profile**—Choose the telephone-event profile from the combination box.



6. Click **OK**.
7. Click **Apply**.

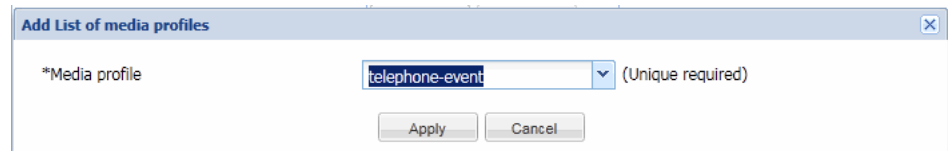
To use DTMF for all IWF translations:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click IWF stack. The IWF data window appears in the content area.
3. Click **Add**. The Add media profile dialog box appears.

4. **Media profile name**—Select the telephone-event media profile and click **OK**.



5. Click **Apply**.

RFC 2833: DTMF Interworking

This section explains the Net-Net SBC's support of transporting Dual Tone Multi-Frequency (DTMF) in Real-Time Transport Protocol (RTP) packets (as described in RFC 2833) to H.245 User Input Indication (UII) or SIP INFO method interworking.

Multimedia devices and applications must exchange user-input DTMF information end-to-end over IP networks. The Net-Net SBC provides the interworking capabilities required to interconnect networks that use different signaling protocols. Also, the Net-Net SBC provides DTMF translation to communicate DTMF across network boundaries.

The Net-Net SBC supports:

- RFC 2833 to H.245 UII translation for H.323-to-H.323 calls, when one side is a version 4 H.323 device requiring RFC-2833 DTMF event packets, and the other side is a pre-version 4 H.323 device that only uses H.245 UII.
- RFC 2833 to H.245 UII or INFO translation of H.323 to SIP (and SIP to H.323) IWF calls, when one side is a version 4 H.323 device requiring RFC 2833 DTMF event packets and the SIP endpoint only supports INFO messages. Or when one side is a pre-version 4 H.323 device that only uses H.245 UII and the SIP endpoint supports RFC-2833 DTMF event packets.

About RFC 2833

RFC 2833 specifies a way of encoding DTMF signaling in RTP streams. It does not encode the audio of the tone itself, instead a signal indicates the tone is being sent. RFC 2833 defines how to carry DTMF events in RTP packets. It defines a payload format for carrying DTMF digits used when a gateway detects DTMF on the incoming messages and sends the RTP payload instead of regular audio packets.

About H.245 UII

H.245 provides a capability exchange functionality to allow the negotiation of capabilities and to identify a set of features common to both endpoints. The media and data flows are organized in logical channels. H.245 provides logical channel signaling to allow logical channel open/close and parameter exchange operations. The H.245 signaling protocol is reliable, which ensures that the DTMF tones will be delivered.

H.245 User Input Indication (UII) plays a key role in all the services that require user interaction. For video messaging, typical uses of UII include selection of user preferences, message recording and retrieval, and typical mailbox management functions. H.245 UII provides two levels of UII, alphanumeric and signal.

About 2833 to H.245 UII Interworking

The Net-Net SBC provides 2833 to H.245-UII interworking by checking 2833-enabled RTP streams for packets matching the payload type number for 2833. It then sends the captured packet to the host for processing and translation to H.245 UII messages. A H.245 UII message received by the Net-Net SBC is translated to 2833 packets and inserted into the appropriate RTP stream.

About DTMF Transfer

DTMF transfer is the communication of DTMF across network boundaries. It is widely used in applications such as interactive voice response (IVR) and calling card applications.

The multiple ways to convey DTMF information for packet-based communications include:

- In-band audio: DTMF digit waveforms are encoded the same as voice packets. This method is unreliable for compressed codecs such as G.729 and G.723
- Out-of-band signaling events:
 - H.245 defines out-of-band signaling events (UII) for transmitting DTMF information. The H.245 signal or H.245 alphanumeric methods separate DTMF digits from the voice stream and send them through the H.245 signaling channel instead of through the RTP channel. The tones are transported in H.245 UII messages.
 - All H.323 version 2 compliant systems are required to support the H.245 alphanumeric method, while support of the H.245 signal method is optional.
 - SIP INFO – uses the SIP INFO method to generate DTMF tones on the telephony call leg. The SIP INFO message is sent along the signaling path of the call. Upon receipt of a SIP INFO message with DTMF content, the gateway generates the specified DTMF tone on the telephony end of the call.
- RTP named telephony events (NTE): uses NTE to relay DTMF tones, which provides a standardized means of transporting DTMF tones in RTP packets according to section 3 of RFC 2833.

Of the three RTP payload formats available, the Net-Net SBC supports RTP NTE. NTE is most widely used for SIP devices but is also supported in H.323 version 4 or higher endpoints.

RFC 2833 defines the format of NTE RTP packets used to transport DTMF digits, hookflash, and other telephony events between two peer endpoints. With the NTE method, the endpoints perform per-call negotiation of the DTMF transfer method. They also negotiate to determine the payload type value for the NTE RTP packets.

The NTE payload takes the place of codec data in a standard RTP packet. The payload type number field of the RTP packet header identifies the contents as 2833 NTE. The payload type number is negotiated per call. The local device sends the payload type number to use for 2833 telephone event packets using a SDP or H.245 Terminal Capability Set (TCS), which tells the other side what payload type number to use when sending the named event packets to the local device. Most devices use payload type number 101 for 2833 packets, although no default is specified in the standard.

The 2833 packet's RTP header also makes use of the timestamp field. Because events often last longer than the 2833 packets sending interval, the timestamp of the first 2833 packet an event represents the beginning reference time for subsequent 2833 packets for that same event. For events that span multiple RTP packets, the RTP

timestamp identifies the beginning of the event. As a result, several RTP packets might carry the same timestamp.

See RFC 2833 and draft-ietf-avt-rfc2833bis-07.txt for more information.

Preferred and Transparent 2833

To support preferred (signaled) 2833 and transparent 2833, the Net-Net SBC provides 2833 detection and generation (if necessary) when the endpoint signals support for 2833.

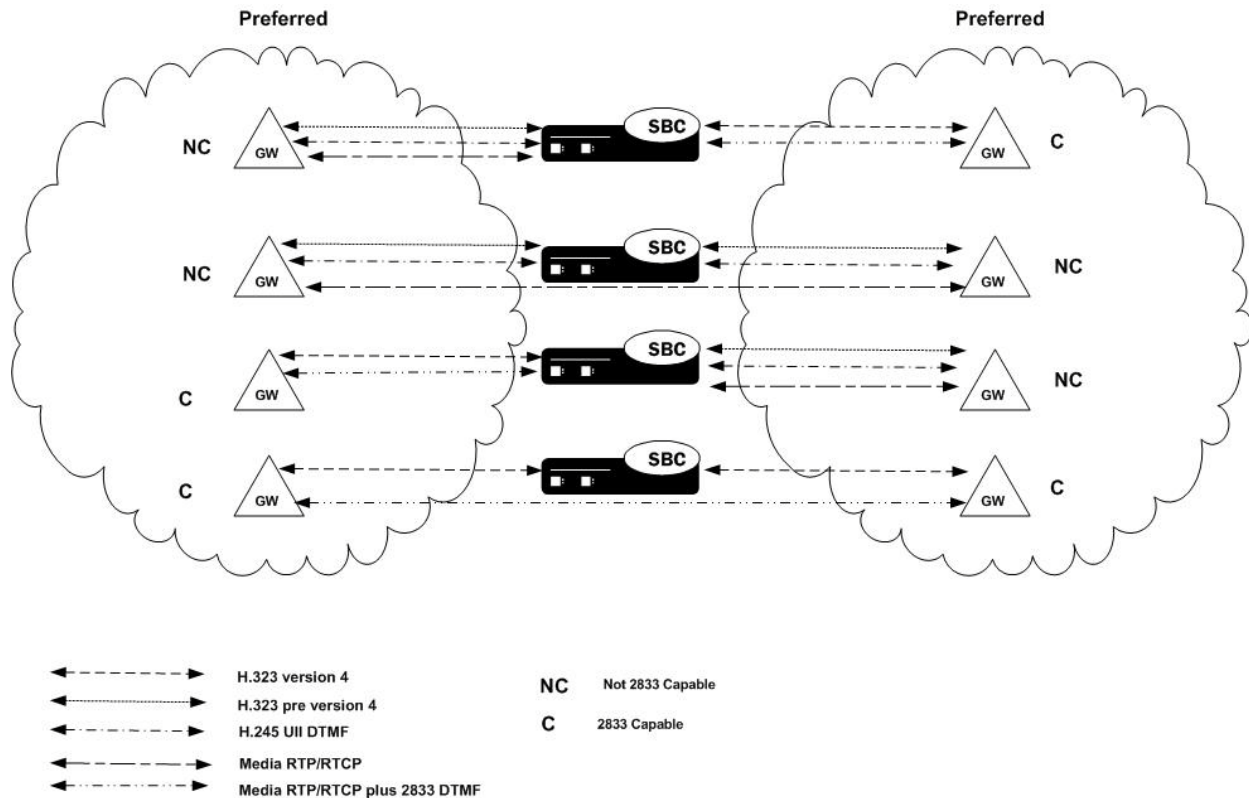
- Preferred: the Net-Net SBC only generates and detects 2833 for endpoints if they negotiate support for 2833 through signaling
- Transparent: the Net-Net SBC behaves as it has prior to this release, offering and answering based on end-to-end signaling and transparently relaying 2833

Preferred 2833 Support

If one side of the call, or a SIP interface, or a session agent, is configured for preferred 2833, the Net-Net SBC only generates and detects 2833 for endpoints if they signal support for 2833. The Net-Net SBC will offer 2833 in the TCS SDP, even if the originating caller did not.

- When the Net-Net SBC manages calls originating from a preferred source going to a preferred target, it:
 - Performs 2833 translation for an endpoint when the originating side requests 2833 but the target does not negotiate 2833
 - Allows 2833 to pass through if the originating side and target of the call are configured as preferred and negotiate 2833
- When the Net-Net SBC manages calls originating from a preferred source going to a transparent target, it:
 - Performs 2833 translation when the originating side requests 2833 but the target is configured as transparent and does not negotiate 2833.

- Allows 2833 to pass through if the originating side and the target of the call are configured as transparent and negotiate 2833. The Net-Net SBC does not perform active translation because both ends support 2833.



If one SIP endpoint does not signal 2833 capability, but the other SIP or H.323 endpoints do, the Net-Net SBC does not perform 2833 translation.

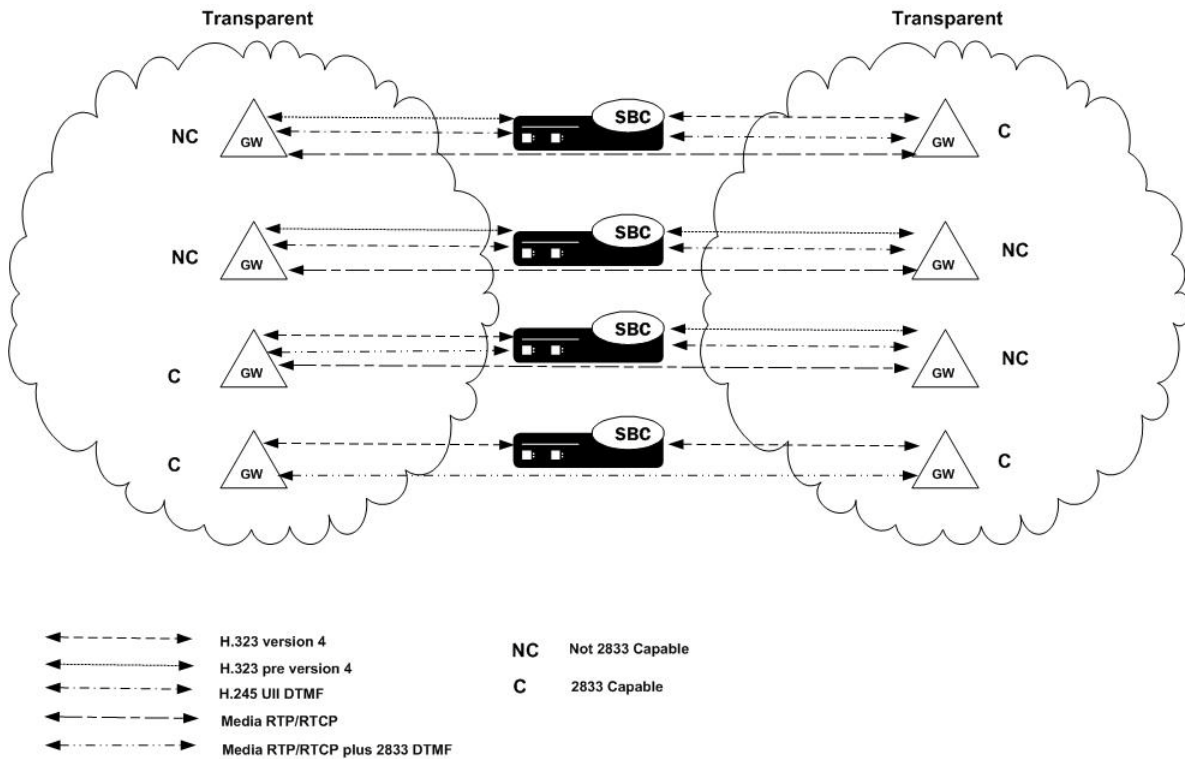
Transparent 2833 Support

The default configuration of the Net-Net SBC for H.323 is transparent 2833. The Net-Net SBC passes on the offered capabilities to the next-hop signaling element. If the next-hop endpoint is for a transparent 2833 target, typical capability negotiation determines the DTMF method. The Net-Net SBC transparently relays the DTMF as it has in previous releases.

With transparent 2833, the Net-Net SBC acts as a typical B2BUA or B2BGW/GK. However when the target of the call is configured as preferred 2833, the Net-Net SBC:

- Relays the 2833 packets if the originating endpoint signals 2833 and the next-hop endpoint for the preferred target signals 2833
- Performs 2833 translation if the originating endpoint does not signal 2833 and the next-hop endpoint for the preferred target does signal 2833

- Does not perform 2833 translation or transparently relay 2833 if the originating endpoint signals 2833 and the next-hop endpoint for the preferred target (or even a transparent 2833 target) does not signal 2833.



Basic RFC 2833 Negotiation Support

If H.323, SIP, or session agents on either side of the call are configured for preferred 2833 support, the Net-Net SBC supports end-to-end signaled negotiation of DTMF on a call-by-call basis. If the calling party is not configured for preferred support but sends 2833, the Net-Net SBC sends 2833 to the next-hop called party. If the calling party sends H.245 signals or alphanumeric UII, the Net-Net SBC sends H.245 signals or alphanumeric UII to the next-hop called party (if it is an H.323 next-hop).

The Net-Net SBC also supports hop-by-hop negotiation of DTMF capability on a call-by-call basis, if the signaling protocols or session agents on either side of the call are configured for preferred 2833 support.

H.323 to H.323 Negotiation

The Net-Net SBC serves as the H.323 called gateway. It answers RFC 2833 audio telephony event capability in the version 4 H.323/H.245 TCS when it receives a call from an H.323 endpoint configured for preferred RFC 2833.

If the Net-Net SBC is the answering device, configured for preferred support, and the calling device sends 2833, the Net-Net SBC accepts the 2833 regardless of the next-hop's DTMF capabilities. The received dynamic RTP payload type is used for detecting 2833 packets, while the response dynamic payload type is used for generating 2833 packets.

The Net-Net SBC supports:

- RFC-2833 audio telephony events in the version 4 H.323/H.245 TCS as the H.323 calling gateway, when the Net-Net SBC calls an H.323 endpoint configured for preferred RFC 2833 support. The Net-Net SBC sends 2833 to the called party regardless of whether the calling party sends it.
- H.245 UII and RFC-2833 packets sent at the same time, to the same endpoint, even if only half of the call is being provided 2833 support by the Net-Net SBC.

If one half of the call supports H.245 UII, and the other half is being provided 2833 translation by the Net-Net SBC, the Net-Net SBC can also forward the H.245 UII it receives to the 2833 endpoint. For example, when the signaling goes through a gatekeeper or third party call control, sending the H.245 UII in the signaling path allows those devices to learn the DTMF digits pressed.

Signal and Alpha Type Support

The Net-Net SBC supports:

- H.245 signal and alpha type UII in the H.323/H.245 TCS as the H.323 calling gateway when the:
 - Net-Net SBC calls an H.323 endpoint configured for transparent 2833 support
 - calling endpoint's target is configured as preferred

If the originating preferred side also sends 2833, the Net-Net SBC forwards it to the transparent side. The Net-Net SBC sends signal and alpha UII support to the called party regardless of whether the calling party sends it, if the call originates from a preferred side to a transparent side.

- H.245 alphanumeric UII for DTMF for H.323 endpoints that do not signal 2833 or contain explicit H.245 UII capability, for stacks configured for transparent 2833 support.

When the other half of the call is an H.323 endpoint of a stack configured for preferred 2833, the Net-Net SBC translates incoming H.245 UII on the transparent side, to 2833 packets on the preferred side, and vice versa. If the other half of the call is an H.323 endpoint of a transparent stack, the Net-Net SBC relays the H.245 UII messages.

- H.245 signal type UII for DTMF for H.323 endpoints that do not signal 2833, but do signal explicit H.245 UII capability, for stacks configured for transparent 2833 support.

When the other half of the call is an H.323 endpoint of a stack configured for preferred 2833, the Net-Net SBC translates incoming H.245 signaled UII on the transparent side, to 2833 packets on the preferred side, and vice versa. If the other half of the call is an H.323 endpoint of a transparent stack, the Net-Net SBC relays the H.245 UII messages if both sides support it.

H.323 to SIP Calls

This section explains DTMF interworking specific to H.323 to SIP calls.

H.323 Endpoints

Because there are different H.323 endpoints based on different versions of H.323, the DTMF can be either be transferred out-of-band as UII or in-band using RFC 2833. Most H.323 endpoints:

- version 4 and above support RFC 2833
- version 2 and pre-version 4 support UII-Signal
- version 1 and pre-version 2 support UII-Alphanumeric

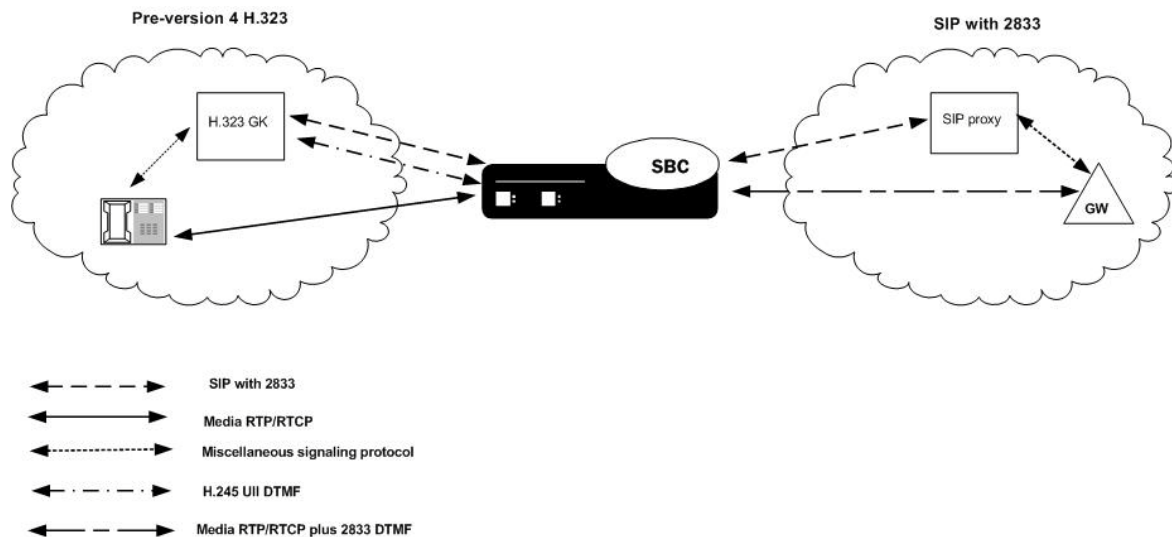
SIP Endpoints

SIP endpoints include those that support:

- RFC 2833
- SIP INFO method

H.323 Non-2833 interworking with SIP

RFC 2833 and the SIP INFO method can be used for conveying DTMF information for SIP based-services. (RFC 2833 is the most widely used.) To provide end-to-end DTMF for SIP devices supporting RFC-2833 interworking with H.323 devices that do not, an RFC 2833 to H.323 UII interworking function is provided.



How H.323 to SIP Calls Work

For H.323 to SIP IWF calls, if 2833-related information is to be sent in the INVITE, the SIP interface of the SIP session agent has to be configured with the rfc2833-mode parameter set to preferred.

The following example shows an INVITE without 2833 in the SDP:

```
Apr  5 04:28:50.073 On 127.0.0.1:5070 sent to 127.0.0.1:5060
INVITE sip:780@192.168.200.6:5060 SIP/2.0
Via: SIP/2.0/UDP
127.0.0.1:5070;branch=z9hG4bKIWF0000gl2018604agg71c0;acme_irealm=external;acme_sa=192.168.1.6
Contact: "rmohamed"<sip:127.0.0.1:5070>
GenericID: 1144211330000000@000825010100
Supported: 100rel^M
From: "msmith"<sip:192.168.200.68:5060>;tag=000000ab00011940
To: <sip:780@192.168.200.6:5060>
Call-ID: 7f00000113ce000000ab000101d0@127.0.0.1
CSeq: 2 INVITE
Content-Length: 225
Content-Type: application/sdp
v=0
o=IWF 3 3 IN IP4 192.168.1.6
s=H323 Call
c=IN IP4 192.168.1.6
t=0 0
m=audio 5214 RTP/AVP 0 18
a=rtpmap:0 PCMU/8000/1
a=rtpmap:18 G729/8000/1
a=fmtp:18 annexb=no
m=video 5216 RTP/AVP 31
a=rtpmap:31 H261/9000/1
```

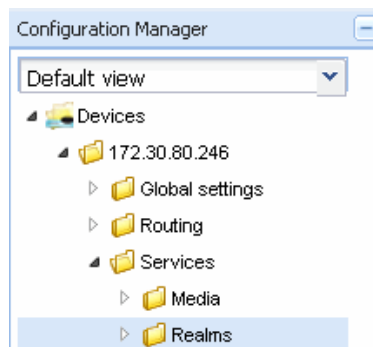
Configuring RFC 2833 for DTMF Interworking

This section describes how to configure DTMF support on your Net-Net SBC from the Net-Net Central.

RFC 2833 Mode for H.323 Stacks

To configure RFC 2833 mode for H.323 stacks:

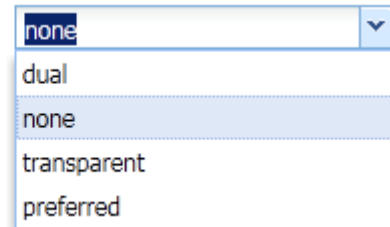
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.

3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab at the top of the content area.
7. **DTMF RFC2833 mode**—Choose the RFC 2833 mode from the combination box.

DTMF RFC2833 mode

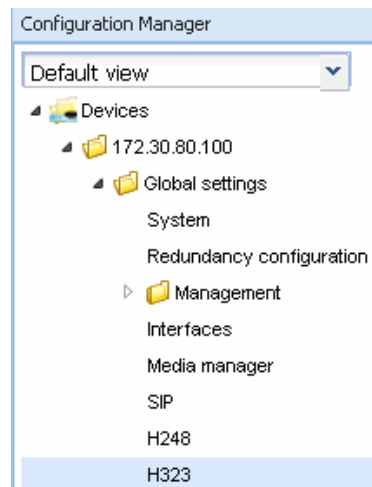


- **transparent**—2833 or UII negotiation is transparent to the Net-Net SBC.
 - **preferred**—The H323 stack uses 2833 for DTMF transfer, which it signals in its TCS. However, the remote H323 endpoint makes the decision. If the endpoint supports 2833, 2833 is used. If not, the H.323 stack reverts back to using UII.
8. Click **Apply**.

RFC 2833 Payload for H.323

To configure the RFC 2833 payload in preferred mode:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. **Payload for RFC2833 packets interworked to UII**—Enter a number that indicates the payload type the Net-Net SBC will use for RFC 2833 packets while interworking 2833 and UII. Value range is 96 to 127, with the default value of 101.

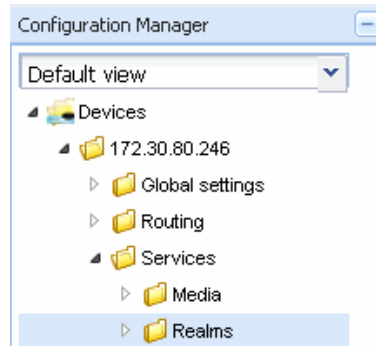
Payload type for RFC2833 packets
interworked to UII

5. Click **Apply**.

Configuring the SIP Interface

You configure the 2833 mode and payload for the SIP interface. You must configure the payload the Net-Net SBC will use for RFC 2833 packets, while interworking 2833 and INFO/UII.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. **Payload type for RFC2833 packets interworked to UII**—Enter a number that indicates the payload type the SIP interface will use for RFC 2833 packets while interworking 2833 and UII. Value range is 0 and 96 to 127.

Payload type for RFC2833 packets
interworked to UII

7. **DTMF RFC2833 mode**—Choose the RFC 2833 mode from the combination box.

DTMF RFC2833 mode

none

▼

dual

none

transparent

preferred

- **transparent**—2833 or UII negotiation is transparent to the Net-Net SBC.
- **preferred**—SIP interface uses 2833 for DTMF transfer.

8. Click **Apply** to continue.

Configuring Session Agents

You configure session agents with:

- Payload type the Net-Net SBC wants to use for RFC 2833 packets while interworking 2833 and UII.

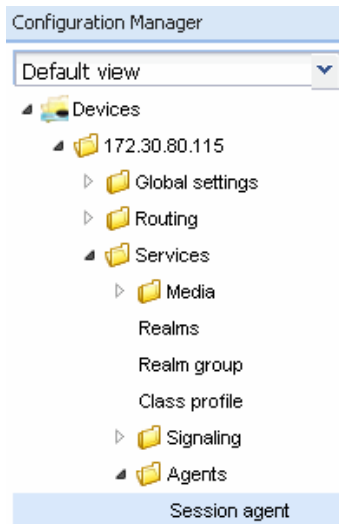
The default value for this attribute is 0. When this value is zero, the global rfc2833-payload configured in the h323-configuration element will be used instead. For SIP session agents, the payload defined in the SIP interface is used, if the SIP interface is configured with the preferred RFC 2833 mode.

- 2833 mode

A value of transparent or preferred for the session agent's 2833 mode will override any configuration in the h323-stack configuration element.

To configure session agents:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the session agent for which you want to configure RFC 2833 to H.245 User UII or SIP INFO method interworking to select it and click **Edit**.
4. Click Translation and manipulation tab at the top of the content area.
5. **Payload type for RFC2833 packets interworked to UII**—Enter a number that indicates the payload type the session agent will use for RFC 2833 packets while interworking 2833 and UII. Value range is 0 and 96 to 127.

Payload type for RFC2833 packets interworked to UII

6. **DTMF RFC2833 mode**—Select the RFC 2833 mode from the combination box.

DTMF RFC2833 mode

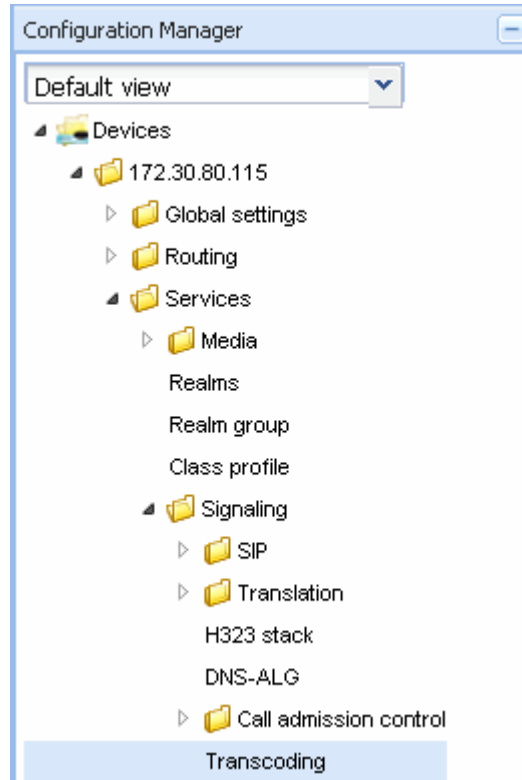
The dropdown menu shows the following options: dual, none (selected), transparent, preferred.

7. Click **Apply**.

Configuring Transcoding Policy

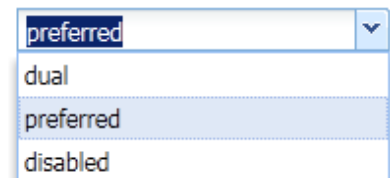
To configure a transcoding policy to support DTMF audio tones:

1. Expand the Configuration Manager slider > Services > Signaling



2. Click Transcoding. The Transcoding table appears in the content area.
3. Click the transcoding policy you want to configure in the Transcoding table and click **Edit**.
4. **Handle DTMF in audio stream**—Click **disabled**, **preferred**, or **dual** in the drop down list. Set this parameter based upon how the Net-Net 9000 should support the conversion of signaling messages, or RFC 2833 to DTMF Audio tones, in the realm where this transcoding policy is active.
 - **disabled**—does not support DTMF audio tones as transcoded in this realm.
 - **preferred**—supports DTMF audio tones as transcoded in this realm.
 - **dual**—supports both transcoded DTMF audio tones and signaling-based DTMF indications if possible.

Handle DTMF in audio stream



5. Click **Apply**.

Payload Type Handling

For calls that require translation between SIP and H.323, the Net-Net SBC supports Real-Time Protocol Named Telephone Events (RTP NTE) (defined in RFC 2833) for telephony events such as transport of DTMF digits and hook flash. RTP NTE is an inband DTMF relay method that uses RTP Named Telephony Event packets to carry

DTMF information instead of voice. Using RPT NTE, endpoints perform per-call negotiation of the DTMF transfer method and negotiate payload type value for the RPT NTE packets.

Although most endpoints use payload type number 101, the RTP payload type formats can become asymmetrical when being interworked between SIP and H.323 because there is no default standard and endpoints use different types.

This means that the payload type negotiated on one side of the call and that ends up being used for the call might not be the same payload type negotiated on the other side. And while certain endpoints handle the asymmetry well, others do not.

In the simplified example of an IWF call initiated in SIP and translated to H.323. The SIP endpoint negotiates the payload type 106 with the Net-Net SBC. Although the H.323 endpoint negotiates payload type 101, the Net-Net SBC returns and the call proceeds using type 106.

You can enable forced symmetric payload type handling so the Net-Net SBC changes the payload type of 2833 packets to avoid using asymmetrical payload types.

How It Works

For SIP session agents and H.323 interfaces (stacks), you can configure an option that forces symmetric payload type use. The Net-Net SBC can detect when the payload types negotiated by the SIP and H.323 endpoint are symmetrical and when they are not. When it detects asymmetrical payload type use, the Net-Net SBC forces the remote endpoint to use the to the RFC 2833 payload type you configure in the SIP interface.

For this feature to work properly, you must set the following SIP interface and the global H.323 configuration parameters with these values:

- **rfc2833 mode**—Set this parameter to **preferred**; the default is **transparent**.
- **rfc2833 payload**—Set this parameter to the payload type you want used for the endpoint. Your entry will be between **96** and **127**, with **101** as the default.

For more information about the Net-Net SBC's RFC 2833 DTMF interworking capabilities, see the *H.323 Feature* chapter of the *Net-Net Configuration Guide*.

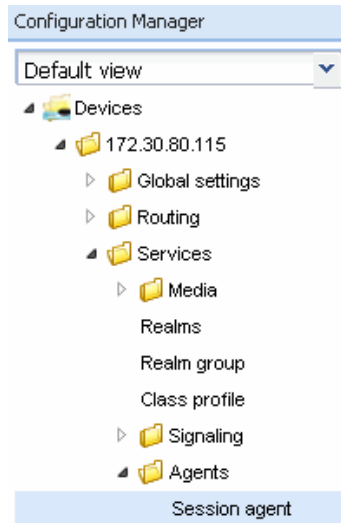
Configuring IWF Payload Type Handling

This section shows how to configure the forced symmetrical payload type option for SIP session agents and H.323 interfaces (stacks).

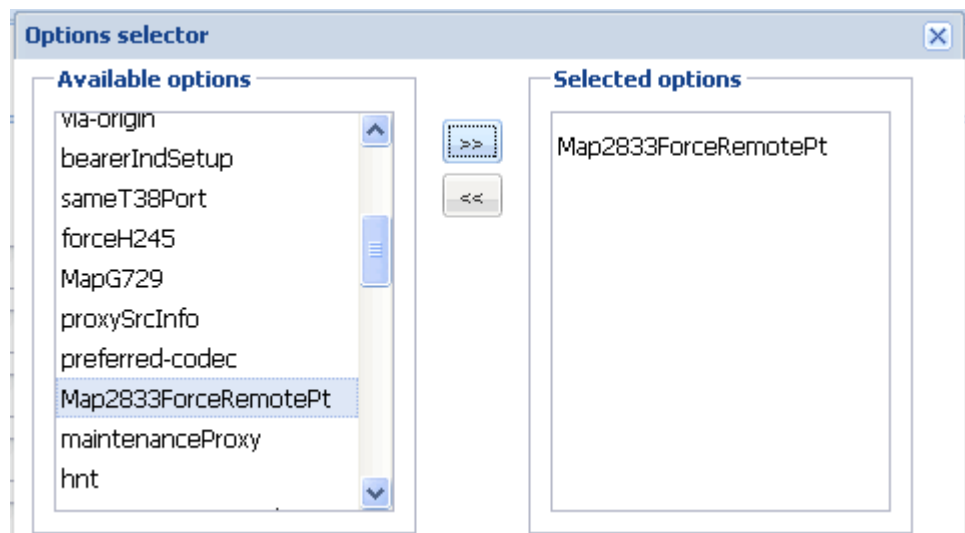
SIP Session Agents

To enable forced symmetric payload type handling for a SIP session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a SIP session agent in the table and click **Edit**.
4. Click the Session agent tab. Scroll to and click **Options**. The Options dialog box appears.
5. Click **Map2833ForceRemotePT** in the Available options list to select it.

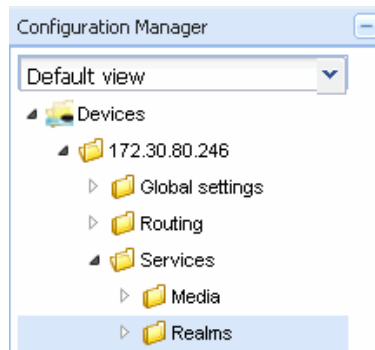


6. Click the right-pointing arrow button to move the options to the Selected options list. The option name appears along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

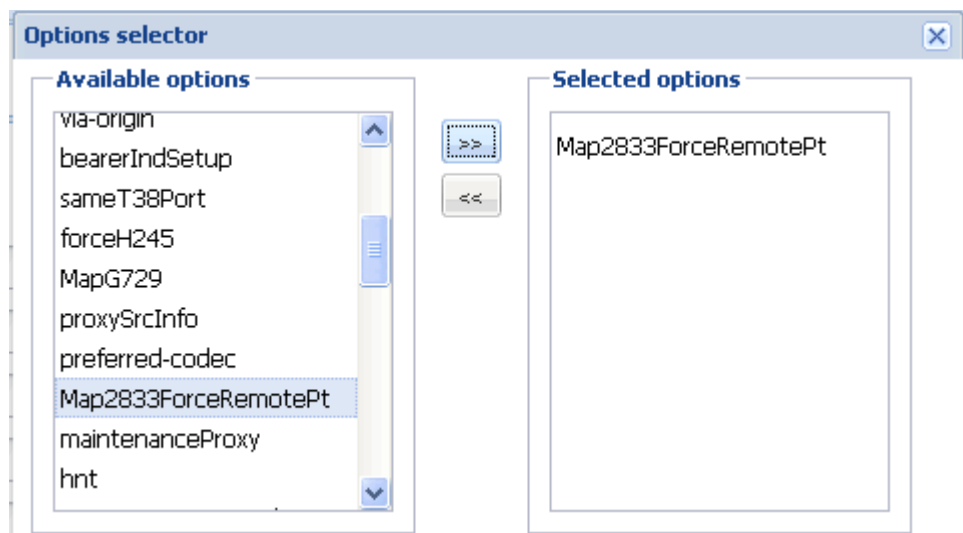
H.323 Interfaces

To enable forced symmetric payload type handling for an H.323 interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click an H.323 realm in the Realm table and click **Edit**. The Edit realm window appears.
4. Click the Services/Protocols tab.
5. Click an H.323 service and click **Edit**. The Edit H323 service window appears.
6. Click the H.323 service mode tab.
7. Scroll to and click **Options**. The Options dialog box appears.
8. Click **Map2833ForceRemotePT** in the Available options list to select it.



9. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
10. Click **OK**.
11. Click **Apply**.

H.245 to 2833 DTMF Interworking Enhancements

H.245 to 2833 and SIP INFO to 2833 DTMF interworking addresses issues experienced where the way the Net-Net SBC timestamps audio RTP packets result in dropped digits and digits with a stutter pattern. These occurrences can cause other

network devices to deem the packets unrecoverable (due to jitter), meaning that they will never render the digit.

The Net-Net SBC offers the following:

- Timestamp is based on the current time—The Net-Net SBC computes the timestamp of the egress 2833 packets using the actual time elapsed in milliseconds since the last RTP packet (rather than incrementing the time by 1 sample). Not only does the Net-Net SBC fill out the timestamp field more accurately, but it also recalculates the checksum
- End-event 2833 messages default behavior—The Net-Net SBC sends three end-event 2833 packets only if the DTMF event is received for:
 - An alphanumeric UII or SIP INFO with no duration
 - A signaled UII or SIP INFO with a duration less than the minimum signal duration (the value you configure using the new media manager configuration **min-signal-duration** option)

For a signaled UII or SIP INFO with a duration greater than the minimum signal duration, the Net-Net SBC sends the initial event packets, any interim packets (if they exist), and the three end packets.

- Configurable duration for the 2833 event—The Net-Net SBC allows you to set the duration of 2833 events using a new **Default duration for RFC2833 packets** parameter (with a 100 millisecond default) on the Media Control tab. In addition, the Net-Net SBC uses this configured value (instead of the duration sent in the signaling message) when it receives an UII or SIP INFO with a duration less than the minimum signal duration. It checks to make sure that the value for the **Default duration for RFC2833 packets** parameter is greater than the minimum signal duration.
- Configurable minimum signal duration value—The Net-Net SBC lets you configure the duration it receives in the UII or SIP INFO for the 2833 event using the **min-signal-duration** option in the media manager configuration. If the duration the Net-Net SBC receives is less than the threshold, it uses the value configured in the **Default duration for RFC2833 packets** parameter.

If you do not configure this option, then there is no signaling duration threshold.

Note: Timestamp changes and duration changes only take effect when you enable the 2833 timestamp (**Adjust timestamp in RFC2833 packets**) is enabled in the Media Control tab.

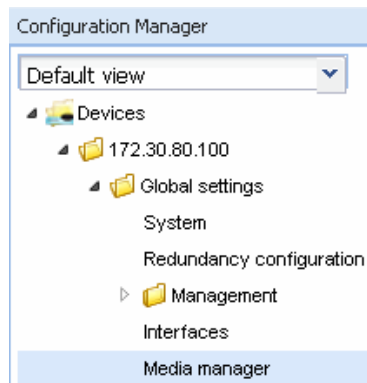
Configuring H.245 UII/SIP INFO-2833 DTMF Interworking

This section shows you how to configure H.245 UII/SIP INFO—2833 DTMF interworking.

Enabling the Timestamp for Current Time

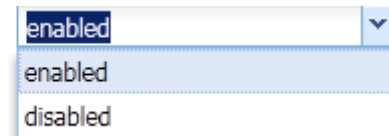
To enable the Net-Net SBC to calculate the timestamp based on the current time:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. **Adjust timestamp in RFC2833 packets**—Select enabled from the combination box to the use of a timestamp value calculated using the actual time elapsed since the last RTP packet.

Adjust timestamp in RFC2833 packets

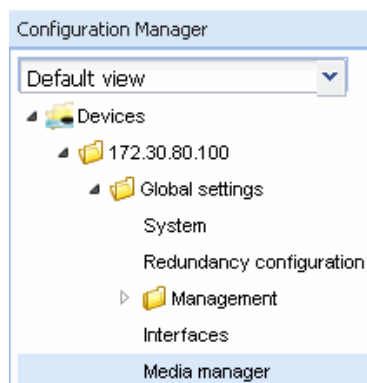


5. Click **Apply**.

Setting the 2833 Event Duration

To configure a duration for the 2833 event:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. **Default duration for RFC2833 packets**—Enter the time value in milliseconds for the Net-Net SBC to use when it receives an alphanumeric UII or a SIP INFO

with no specified duration; then the three end-event packets are sent. The entries can range from 50 to 5000 milliseconds.

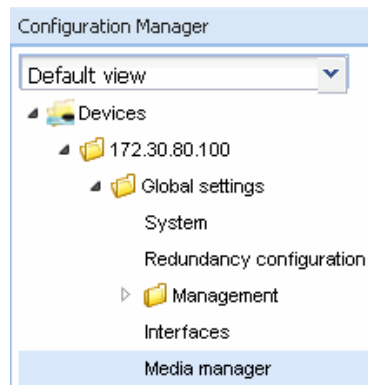
Default duration for RFC2833 packets

5. Click **Apply**.

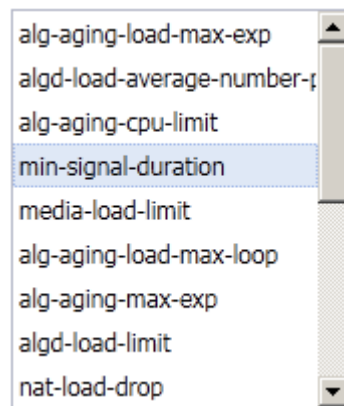
Setting the Minimum Signal Duration

To configure the minimum signal duration value:

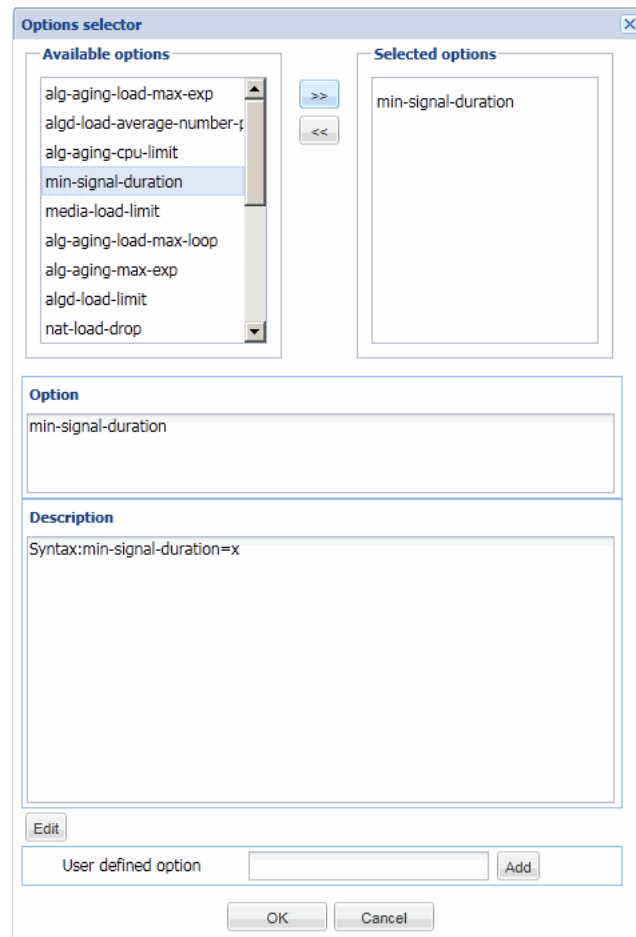
1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to the Options area.
5. Click **Options**. The Options dialog box appears.
6. Click **min-signal-duration** in the Available options list to select it.



7. Click  to move the option to the Selected options list.



8. **min-signal-duration**—Edit the min-signal-route textbox to enter the value in milliseconds you want to use for the threshold.

`min-signal-duration=200`

9. Click OK.
10. Click Apply.

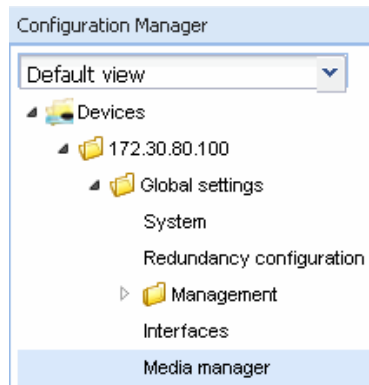
Additional H.245 to 2833 DTMF Interworking

You can configure your Net-Net SBC to generate either the entire start-interim-end RFC 2833 packet sequence, or only the last three end 2833 packets for non-signaled digit events.

Configuring RFC 2833 End Packets Only for Non-Signaled Digits Events

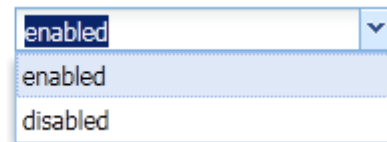
To send only the last three end 2833 packets for non-signaled digits events:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to the **Enable sending only end packet for non-signaled events** checkbox.
5. **Enable sending only end packet for non-signaled events**—Select enabled from the combination box to enable only the last three end 2833 packets for non-signaled digits events.

Enable sending only end packet for non-signaled events



6. Click **Apply**.

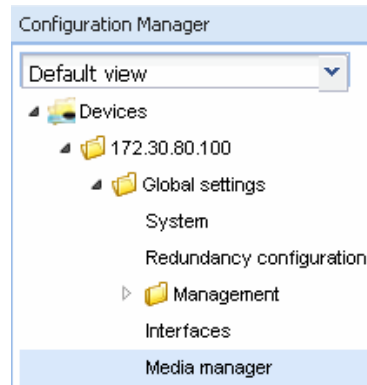
DTMF Transparency for IWF

In certain vendors' implementations of DTMF during SIP/H.323 IWF, there have been discrepancies between the RFC 2833 and UII/INFO negotiations and the type of messages actually sent. You can configure the Net-Net SBC to always send the correctly negotiated protocols, regardless of the type of messages it may be receiving.

Configuring DTMF Transparency for IWF

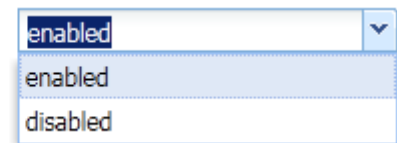
To enable DTMF transparency for SIP/H.323 IWF:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to Accept non rfc2833 events.
5. **Accept non rfc2833 events**—Select enabled from the combination box.

Accept non RFC2833 events



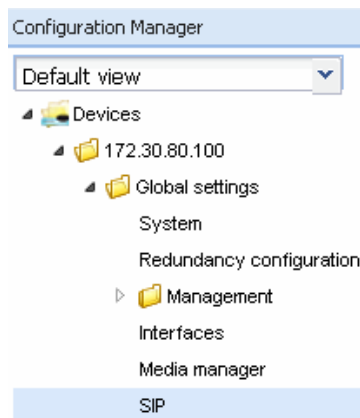
6. Click **Apply**.

Configuring the P-dual-info Header Appearance

Customizing the P-Dual-Info header is performed globally from the SIP configuration.

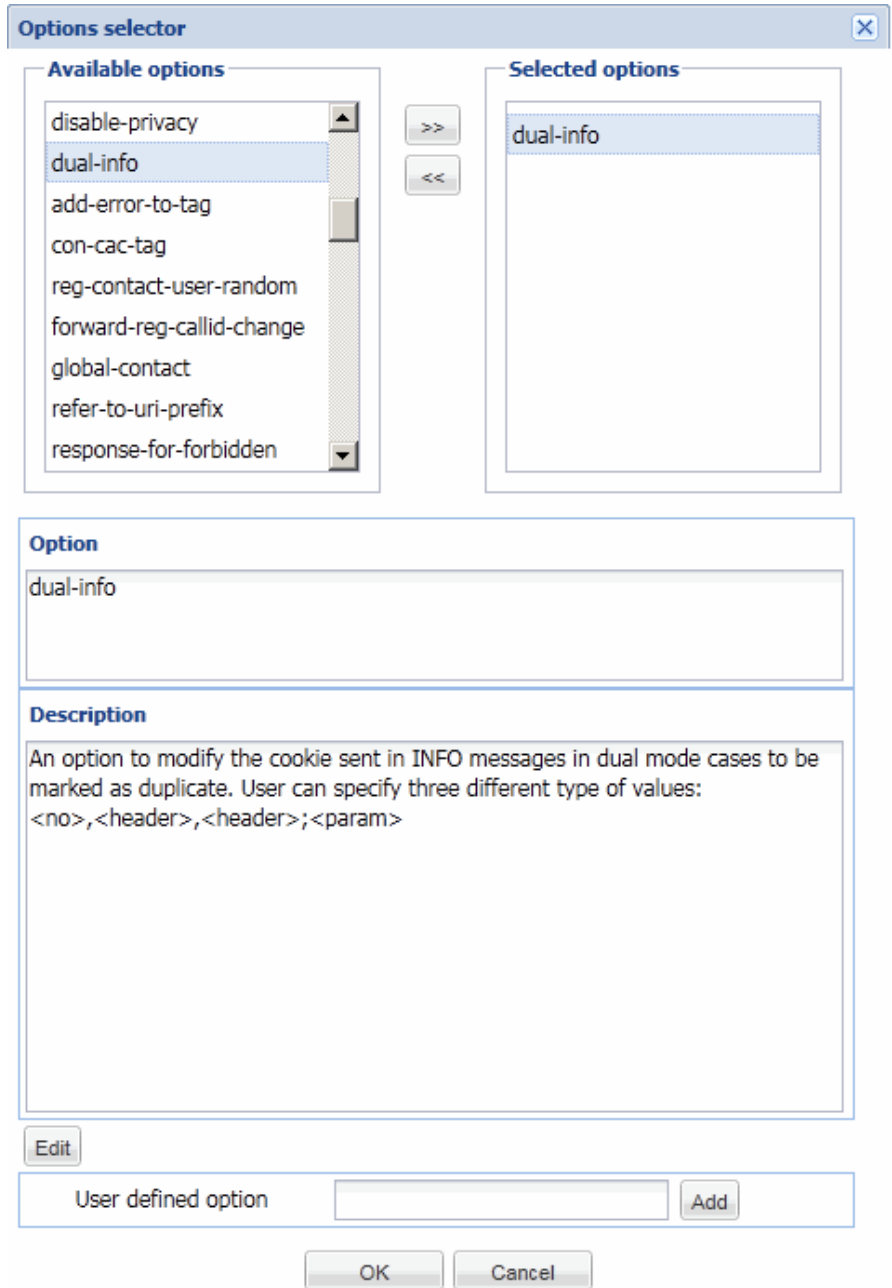
To configure how the P-dual-info header appears:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The Options window appears.
5. Click **dual-info** from the Available options list to select it.

6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



The **Options selector** dialog box is shown. It has a title bar with a close button. Inside, there are two main sections: **Available options** and **Selected options**. The **Available options** section contains a list box with the following items: disable-privacy, dual-info (highlighted), add-error-to-tag, con-cac-tag, reg-contact-user-random, forward-reg-callid-change, global-contact, refer-to-uri-prefix, and response-for-forbidden. To the right of this list are two buttons: **>>** and **<<**. The **Selected options** section contains a list box with the item **dual-info**. Below these sections, there is a detailed view for the selected option. It has a label **Option** with the text **dual-info** in a text field. Below that is a label **Description** with a text area containing the text: "An option to modify the cookie sent in INFO messages in dual mode cases to be marked as duplicate. User can specify three different type of values: <no>, <header>, <header>;<param>". Below the description is an **Edit** button. At the bottom of the dialog, there is a section for **User defined option** with a text input field and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

7. Click **OK**.
8. Click **Apply**.

SIP Tel URI Support

The Net-Net SBC maps H.323 addresses to either SIP URIs or Tel URIs. You can configure the Net-Net SBC to include Tel URIs in the following SIP headers for calls that require the IWF:

- Request Line

- To
- From

When Tel URI support is not used on a Net-Net SBC performing IWF translations, the SIP INVITE is formatted like it is in the following example. This example uses 192.168.5.5 as the external proxy address, or the next hop (as configured in the local policy).

```
INVITE sip:602@192.168.5.5:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.5.58:5060;branch=z9hG4bKIWF0aqq001g11a7kos4g0
Contact: <sip:603@192.168.5.58:5060>
From: <sip:603@192.168.5.58:5060>;tag=4069ac210018a0
To: <sip:602@192.168.5.5:5060>
```

In the example above, the session needs to be routed to another SIP proxy that can resolve an E.164 number to a SIP address. However, the next SIP proxy must be informed that the message will be routed based on the included E.164 number; the SIP address of the Request URI does not have a routable SIP address. To devise a routable address, the Request URI must be reconstructed as a Tel URI.

Without Tel URI support configured, the terminating SIP user would be required to have an address of 602@192.168.5.5, where the IP address portion is the same as the address for the proxy. If it were not the same, then the session would terminate at the proxy. However, the proxy would be unable to handle the session because the SIP address it received would be unknown/unroutable.

Because it is not desirable to have an IP address be the user-identity and rely on the configuration of the IP network, the SIP INVITE generated by the Net-Net SBC and sent to the proxy must have the following format if it is sent to an H.323 entity.

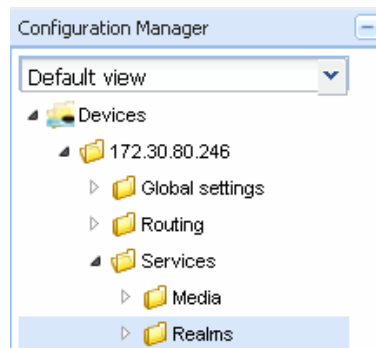
```
INVITE tel:2345 SIP/2.0
Via: SIP/2.0/UDP 192.168.5.52:5060;branch=z9hG4bKIWFaqq00cobgf9so10o0
Contact: <sip:1234@192.168.5.58:5060>
From: <tel:1234>;tag=4069ac35000c5ff8
To: <tel:2345>
Call-ID:7f0000113ce4069ac35000c5440
CSeq: 1 INVITE
Content-Length: 155
Content-Type: application/sdp
```

Configuring SIP Tel URI Support

This section describes how to configure SIP Tel URI support on your Net-Net SBC from the Net-Net Central.

To configure SIP Tel URI support:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to URI - To, From, and request headers.
7. **Scheme**—Select enabled from the combination box.



8. Click **Apply**.

IWF Inband Tone Option

This option enables the Net-Net SBC to send a progress indicator (PI) =8 in an H.225 message when an SDP is received in a provisional message. In effect, this option sends network announcements inband. It is also applicable because in some networks H.323 endpoints support early H.245.

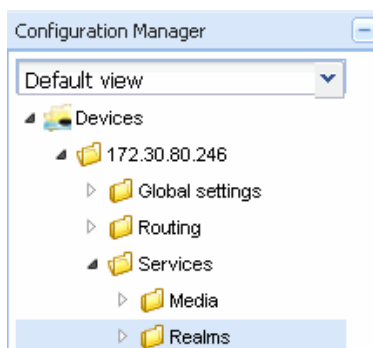
The H.323 inband tone option is enabled by adding the **inbandTone** as an option in a configured H.323 stack.

When this option is not used, the ringtone is generated locally (NO PI=8 in PROGRESS OR ALERTING) is the default behavior.

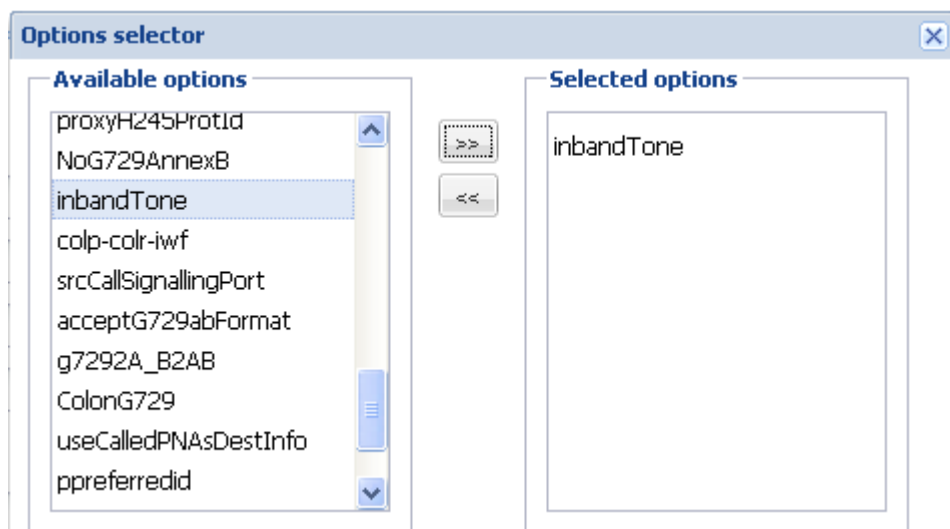
Configuring IWF Inband Tone Option

To configure the IWF inband tone option:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 interface service in the table and click **Edit**.
6. Click the H.323 service mode tab.
7. Scroll to the Options for H323 Service section and click **Options**. The Options dialog box appears.
8. Click **inbandTone** in the Available options list to select it.



9. Click  to move the inbandTone option to the opposite list. The options name and description appear in the middle of the screen.
10. Click **OK**.
11. Click **Apply**.

RFC 3326 Support

This section explains the Net-Net SBC's ability to map Q.850 cause values with SIP responses for calls that require IWF.

RFC 3326 defines a header that might be included in any in-dialogue request. This reason header includes cause values that are defined as either a SIP response code or ITU-T Q.850 cause values. You can configure the Net-Net SBC to support sending and receiving RFC 3326 in SIP messages for:

- Mapping H.323 Q.850 cause values to SIP responses with reason header and cause value
- Mapping SIP response messages and RFC 3326 reason header and cause
- Locally generated SIP response with RFC 3326 reason header and cause

As specified in RFC 3326, the Net-Net SBC sends SIP responses to the softswitch that contain the received Q.850 cause code and the reason.

Though the Net-Net SBC can generate RFC 3326 headers, the default behavior for this feature is disabled. Furthermore, the Net-Net SBC can receive and pass SIP error messages (4xx, 5xx, and 6xx) that contain the SIP reason header with a Q.850 cause code and reason (as specified in RFC 3326). If the Net-Net SBC receives an error message without the Reason header, then the Net-Net SBC is not required to insert one.

In calls that require IWF, the Q.850 cause generated in the SIP response are the same as the cause received in the following H.225 messages: Disconnect, Progress, Release, Release Complete, Resume Reject, Status, and Suspend Reject. In addition, the Q.850 cause codes that the Net-Net SBC receives in RFC 3326 headers are passed to the H.323 part of the call unmodified; the H.323 call leg uses this cause code for releasing the call.

For interworking calls between SIP and H.323, you can configure mappings for:

- SIP status codes to Q.850 values
- Particular Q.850 cause codes to SIP status codes

If it cannot find the appropriate mapping, then the Net-Net SBC uses default mappings defined in the Default Mappings table below.

The following describes how the Net-Net SBC handles different IWF call scenarios:

- SIP request containing a Reason header—When it receives a request containing a Reason header, the Net-Net SBC determines if the request is a SIP BYE or SIP CANCEL message. RFC 3326 states that the Reason header is mainly used for these types of requests. If there is a Reason header and it contains the Q.850 cause value, then the Net-Net SBC releases the call on the H.323 side using the specified cause value.
- SIP response—When it receives the error response to an initial SIP INVITE, the Net-Net SBC uses its SIP-Q.850 map to determine the Q.850 that it will use to release the call. If there is not a map entry, then the Net-Net SBC uses the default mappings shown in the Default Mappings table.
- Active call released from the H.323 side—If an active call is released from the H.323 side, the Net-Net SBC checks the outgoing realm (the SIP side) to see if the addition of the Reason header is enabled. If it is, then the Net-Net SBC adds the Reason header in the SIP BYE request with the Q.850 value it received from the H.323 side.

- Error during setup of the call on the H.323 side—In the event of an error during setup on the H.323 side of the call, the Net-Net SBC needs to send:
 - An error response, if this is a SIP to H.323 call
 - A SIP CANCEL, if this is a H.323 to SIP call and the H.323 side hangs up before the call is answered on the SIP side

In this case, the Net-Net SBC checks to see if adding the Reason header is enabled in the IWF configuration. If it is, then the Net-Net SBC adds the Reason header with the Q.850 cause value it received from the H.323 side.

- Call released due to a Net-Net SBC error—If the call is released due a Net-Net SBC error and adding the Reason header is enabled in the IWF configuration, the error response to the initial INVITE contains the Reason header. The Net-Net SBC checks the SIP to Q.850 map configurations to determine whether or not the SIP error response code it is generating is configured. If it is, then the Net-Net SBC maps according to the configuration. If it is not, the Net-Net SBC derives cause mapping from the default table.

Like the configuration for SIP-only calls that enable this feature, you can set a parameter in the IWF configuration that enables adding the Reason header in the SIP requests or responses.

Default Mappings

This table defines the default mappings the Net-Net SBC uses when it cannot locate an appropriate entry that you have configured.

Q.850 Cause Value		SIP Status		Comments
1	Unallocated number	404	Not found	
2	No route to specified transit network	404	Not found	
3	No route destination	404	Not found	
16	Normal calling clearing		BYE message	A call clearing BYE message containing cause value 16 normally results in the sending of a SIP BYE or CANCEL request. However, if a SIP response is to be sent to the INVITE request, the default response code should be used.
17	User busy	486	Busy here	
18	No user responding	408	Request timeout	
19	No answer from the user	480	Temporarily unavailable	
20	Subscriber absent	480	Temporarily unavailable	

Q.850 Cause Value		SIP Status		Comments
21	Call rejected	603	Decline (if location filed in Cause information element indicates user; otherwise 403 Forbidden is used)	
22	Number changed	301	Moved permanently (if information in diagnostic field of Cause information element is suitable for generating SIP Contact header; otherwise 410 Gone is used)	
23	Redirection to new destination	410	Gone	
25	Exchange routing error	483	Too many hops	
27	Destination out of order	502	Bad gateway	
28	Address incomplete	484	Address incomplete	
29	Facility rejected	501	Not implemented	
31	Normal, unspecified	480	Temporarily unavailable	
34	No circuit, channel unavailable	503	Service unavailable	
38	Network out of order	503	Service unavailable	
41	Temporary failure	503	Service unavailable	
42	Switching equipment congestion	503	Service unavailable	
47	Resource unavailable unspecified	503	Service unavailable	
55	Incoming calls barred with CUG	403	Forbidden	
57	Bearer capability not authorized	403	Forbidden	
58	Bearer capability not presently available	503	Service unavailable	
65	Bearer capability not implemented	488	Not acceptable here	
69	Requested facility not implemented	501	Not implemented	

Q.850 Cause Value		SIP Status		Comments
70	Only restricted digital information available	488	Not acceptable here	
79	Service or option not implemented, unspecified	501	Not implemented	
87	User not member of CUG	403	Forbidden	
88	Incompatible destination	503	Service unavailable	
102	Recovery on timer expiry	504	Server time-out	

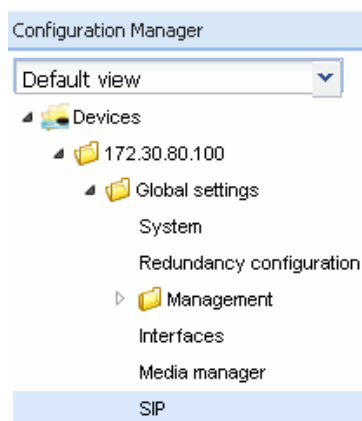
Configuring RFC 3326 Support

You can use Net-Net Central to configure a Q.850 cause to a SIP status with reason mapping.

Adding the Reason Header

To enable the Net-Net SBC to add the Reason header for calls that require IWF:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **Reason headers in SIP responses**—Select enabled from the combination box.

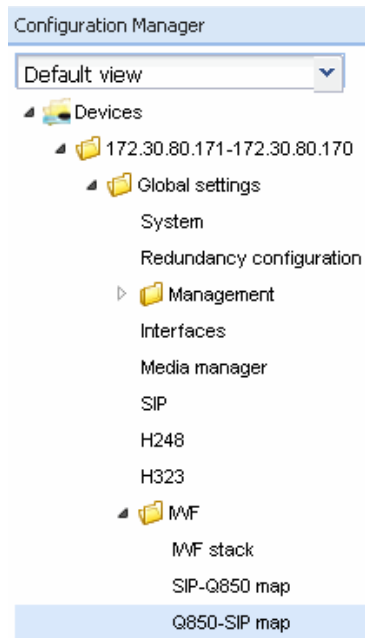
Reason headers in SIP responses

5. Click **Apply**.

SIP Status to Q.850

To configure a SIP status to Q.850 Reason with cause mapping:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click SIP-Q850 Map.
3. Click **Add** to add the configuration. The SIP > H.323 calls table appears in the content area.
4. **SIP status code**—Set the SIP response code that you want to map to a particular Q.850 cause code and reason. The valid range for values is 100-699.
5. **Q850 cause code**—Set the Q.850 cause code that you want to map to the SIP response code that you set in step 5.
6. **Q850 reason**—Set the Q.850 reason corresponding to the Q.850 cause code. There is no default. If your value has spaces between characters, then your entry must be surrounded by quotation marks.

Add SIP->H323 calls		
*SIP status code	486	(Unique required. Range:100..699. Default:100)
*Q850 cause code	17	(Range:0..999999999. Default:0)
*Q850 reason	User Busy	
Apply Cancel		

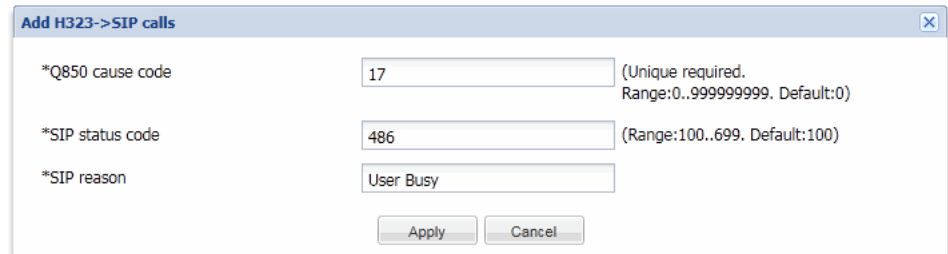
7. Click **Apply**. The entry appears in the SIP->H.323 calls table.
8. Repeat this process to create the number of local response map entries that you need.
9. Click **Apply**.

Q.850 Cause to SIP Status

To configure a Q.850 cause to a SIP status with reason mapping:

1. With the SIP-H.323 IWF tab displayed in the Net-Net Central display pane, scroll to the SIP status to Q.850 cause mappings section.

2. Click **Add**. The Add Q.850 SIP Map Entry dialog box appears.
3. **Q850 cause code**—Enter the Q.850 cause code that you want to map to a SIP status with reason.
4. **SIP status code**—Enter the SIP status code to which you want to map the Q.850 cause that you set in step 5. The valid range for a value is 100-699.
5. **SIP reason**—Enter the SIP reason that you want to use with the SIP response code. If your value has spaces between characters, then your entry must be surrounded by quotation marks.



The image shows a dialog box titled "Add H323->SIP calls" with a close button (X) in the top right corner. It contains three input fields with labels and validation text:

Field Label	Value	Validation Text
*Q850 cause code	17	(Unique required. Range:0..999999999. Default:0)
*SIP status code	486	(Range:100..699. Default:100)
*SIP reason	User Busy	

At the bottom of the dialog box are two buttons: "Apply" and "Cancel".

6. Click **Apply**. The new entry appears in the H323->SIP calls table.
7. Repeat this process to create the number of local response map entries that you need.
8. Click **Apply**.

IWF Privacy: Caller Privacy on Unsecure Networks

This feature enables bi-directional SIP/H.323 IWF support for CPID hiding by using the presentation indicators in the Calling Party Number information element for H.323 signaling, and RFC 3325-based privacy support for SIP signaling. It lets the Net-Net SBC insert the P-Asserted-Identity and the Privacy header in the INVITE when the presentation indicator is set to **restricted**.

The presence, or absence, of P-Asserted-Identity and Privacy headers in the SIP INVITE informs the remote SIP proxy or endpoint to either block or advertise the CPID.

About the Presentation Indicator

When address information represents a telephone number, the relevant information can appear in the Calling Party Number information element (IE). This IE contains the caller's number, information about the number, and presentation and screening indicators found in octet 3a. In order to prevent a calling party number to be passed through, the presentation indicator parameter (octet 3a) in the Calling Party IE must be set to a value other than 00.

In a H.323 to SIP IWF call, octet 3a in the Q.931 message indicates the caller's preference for CPID restriction. If bits 7 and 6 are set to (0 1), the presentation is restricted and the outbound SIP INVITE from the IWF stack must be constructed as such.

H.323 to SIP IWF Call

When the presentation indicator in the calling party IE is set to restricted, the INVITE's From and Contact headers sent from to sipd will be modified according to RFC 3325. When the Net-Net SBC receives calls initiated as H.323, it will recognize the caller's presentation bits as defined in Q.931 and use that information to construct a SIP INVITE in accordance with the user's indicated preference.

- Inclusion of a P-Asserted-Identity header in the INVITE, containing the calling party's CPID and the Net-Net SBC's IP address, constructed as a SIP URI (same mechanism used to construct the From-URI today).
- Addition of a Privacy header with its value set to "id". This addition indicates to the upstream proxies and gateways that the caller address is to be hidden.

The sipd will either proxy or strip these headers according to RFC 3325, depending on the SIP interface and SIP session agent configurations.

Example 1: SETUP Sent from h323d to Remote H.323 Endpoints

```
Q.931
Protocol discriminator: Q.931
Call reference value length: 2
Call reference flag: Message sent from originating side
Call reference value: 2F62
Message type: SETUP (0x05)
Bearer capability
Information element: Bearer capability
Length: 3
...0 1000 = Information transfer capability: Unrestricted digital
information (0x08)
.00. .... = Coding standard: ITU-T standardized coding (0x00)
```

```

1... .... = Extension indicator: last octet
...1 0011 = Information transfer rate: 384 kbit/s (0x13)
.00. .... = Transfer mode: Circuit mode (0x00)
1... .... = Extension indicator: last octet
...0 0101 = User information layer 1 protocol: Recommendation H.221 and
H.242 (0x05)
1... .... = Extension indicator: last octet
Display 'rmohamed\000'
Information element: Display
Length: 9
Display information: rmohamed\000
Calling party number
Information element: Calling party number
Length: 2
.... 0000 = Numbering plan: Unknown (0x00)
.000 .... = Number type: Unknown (0x00)
0... .... = Extension indicator: information continues through the next
octet
.... ..00 = Screening indicator: User-provided, not screened (0x00)
.01. .... = Presentation indicator: Presentation restricted (0x01)
1... .... = Extension indicator: last octet

```

Example 2: INVITE from h323d to sipd

The two new headers will be stripped by the sipd when the INVITE is sent to a untrusted SIP proxy or endpoint and will be proxied over to a trusted SIP proxy or end point.

```

INVITE sip:780@192.168.200.6:5060;acme_realm=internal SIP/2.0
Via: SIP/2.0/UDP
127.0.0.1:5070;branch=z9hG4bKIWF00000510d031s9kou5c0;acme_irealm=external
Contact: "Anonymous"<sip:anonymous@127.0.0.1:5070
GenericID: 7400000@000825010100
Supported: 100rel
From: "Anonymous"<sip:anonymous@anonymous.invalid>;tag=0000004a000d8cc0
To: <sip:780@192.168.200.6:5060
Call-ID: 7f00000113ce0000004a000d88d8@127.0.0.1
CSeq: 2 INVITE
P-Asserted-Identity: "rmohamed"<sip:42343@192.168.200.68:5060>
Privacy: id
Content-Length: 175
Content-Type: application/sdp

v=0
o=IWF 3 3 IN IP4 192.168.1.6
s=H323 Call
c=IN IP4 192.168.1.6
t=0 0
m=audio 5666 RTP/AVP 0 101 18
a=rtpmap:0 PCMU/8000/1

```

```

a=rtpmap:101 telephone-event/8000/1
a=fmtp:101 0-15
a=rtpmap:18 G729/8000/1
a=fmtp:18 annexb=no
m=video 5668 RTP/AVP 31
a=rtpmap:31 H261/9000/1

```

SIP to H.323 IWF Call

For a SIP to H.323 call, the Net-Net SBC must recognize the caller's Privacy request and set the presentation bits accordingly when constructing the outbound RAS/SETUP message. It must check SIP calls for the Privacy header (with value set to "id"). If this header is present, the SETUP's octet 3a's presentation bits must be set to restricted.

The Net-Net SBC does not support any other value for the Privacy header. For those calls, the SETUP will not include a presentation indicator.

Example: INVITE from SIP End Point to sipd

```

Apr 21 08:50:38.786 On [0:0]192.168.200.68:5060 received from
192.168.200.6:5062

```

```

INVITE sip:800@192.168.200.68:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.200.6:5062
From: anonymous <sip:anonymous@192.168.200.6:5062>;tag=1
To: sut <sip:800@192.168.200.68:5060
P-Asserted-Identity: sipp <sip:7789@192.168.200.6:5062
Privacy: id
Call-ID: 1.1688.192.168.200.6@sipp.call.id
Cseq: 1 INVITE
Contact: sip:anonymous@192.168.200.6:5062
Max-Forwards: 70
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 136

```

```

v=0
o=user1 53655765 2353687637 IN IP4 127.0.0.1
s=-
t=0 0
c=IN IP4 127.0.0.1
m=audio 10000 RTP/AVP 0
a=rtpmap:0 PCMU/8000

```

Sample INVITE from sipd to h323d

```

Apr 21 08:50:38.807 On 127.0.0.1:5070 received from 127.0.0.1:5060
INVITE sip:800@127.0.0.1:5070;acme_sag=sag1;acme_irealm=internal SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:5060;branch=z9hG4bK0804o700c0f0t9gpj0g0.1
From: anonymous <sip:anonymous@192.168.200.6:5062>;tag=SDm8kvc01-1
To: sut <sip:800@192.168.200.68:5060
P-Asserted-Identity: sipp <sip:7789@192.168.200.6:5062

```

Privacy: id

Call-ID: SDm8kvc01-083221d8c0fa33f71ae85dd6ed0e4ea4-06ahc21
 Cseq: 1 INVITE
 Contact: <sip:anonymous@192.168.200.68:5060;transport=udp>
 Max-Forwards: 69
 Subject: Performance Test
 Content-Type: application/sdp
 Content-Length: 136
 GenericID: 9883100005@000825010100

v=0

o=user1 53655765 2353687637 IN IP4 127.0.0.1

s=-

t=0 0

c=IN IP4 127.0.0.1

m=audio 10000 RTP/AVP 0

a=rtpmap:0 PCMU/8000

Sample SETUP sent from h323d to remote H323 EP

Q.931

Protocol discriminator: Q.931

Call reference value length: 2

Call reference flag: Message sent from originating side

Call reference value: 664D

Message type: SETUP (0x05)

Bearer capability

Information element: Bearer capability

Length: 3

...1 0000 = Information transfer capability: 3.1 kHz audio (0x10)

.00. = Coding standard: ITU-T standardized coding (0x00)

1... = Extension indicator: last octet

...1 0000 = Information transfer rate: 64 kbit/s (0x10)

.00. = Transfer mode: Circuit mode (0x00)

1... = Extension indicator: last octet

...0 0011 = User information layer 1 protocol: Recommendation G.711

A-law (0x03)

1... = Extension indicator: last octet

Display 'anonymous'

Information element: Display

Length: 9

Display information: anonymous

Calling party number

Information element: Calling party number

Length: 2

.... 0000 = Numbering plan: Unknown (0x00)

.000 = Number type: Unknown (0x00)

```

0... .... = Extension indicator: information continues through the
next octet
.... ..00 = Screening indicator: User-provided, not screened (0x00)
.01. .... = Presentation indicator: Presentation restricted (0x01)
1... .... = Extension indicator: last octet

```

IWF Privacy: Caller Privacy on Secure Connections

In prior releases, when the H.323 endpoint sends a SETUP with presentation indicator set to **allowed**, the Net-Net SBC does not insert the P-Asserted-Identity in the INVITE. The SIP INVITE needs the P-Asserted-Identity header to support calling line identification presentation (CLIP) to calling line identification restriction (CLIR) in an IP multimedia subsystem (IMS) solution. This feature lets the Net-Net SBC insert the P-Asserted-Identity in the INVITE when the presentation indicator is set to **allowed**.

- CLIP is a service provided to the called party that allows the display of the calling number (caller ID). The user-provided calling number must be transported from the caller to the called party.
- CLIR is a service provided to the calling party that lets it indicate whether or not the calling number is to be displayed to the called party. It sets a calling number presentation indicator to **allowed** or **restricted**. Regulations require that network administrations remove the calling number before it is sent to the called party, if the calling party has so requested.

H.323 to SIP IWF

When the Net-Net SBC translates incoming H.323 messages to SIP on a secure connection (which means the Net-Net SBC can rely on the data sent from the originator); it will translate the information in the H.323 messages into SIP messages as detailed in the following sections.

Calls with Presentation Allowed

When the Net-Net SBC receives a SETUP from the H.323 domain where presentation is allowed, it generates an INVITE to the SIP domain with the following header. (Presentation is allowed when the calling party's information element presentation indicator (octet 3a) equals 00.)

- P-Asserted-ID: the userpart should be derived from the Calling Party Number Information Element digits.

H.323 to SIP

When h323d receives a SETUP with the calling party's information element presentation indicator set to **allowed**, the Net-Net SBC will add the P-Asserted-Identity header to the INVITE. The P-Asserted-Identity is very similar to the FROM header, except for the tag.

Sample SETUP sent from h323d to Remote H323 Endpoints

```

Q.931
Protocol discriminator: Q.931
Call reference value length: 2
Call reference flag: Message sent from originating side
Call reference value: 2F62
Message type: SETUP (0x05)
Bearer capability
Information element: Bearer capability
Length: 3
...0 1000 = Information transfer capability: Unrestricted digital
information (0x08)
.00. .... = Coding standard: ITU-T standardized coding (0x00)
1... .... = Extension indicator: last octet
...1 0011 = Information transfer rate: 384 kbit/s (0x13)
.00. .... = Transfer mode: Circuit mode (0x00)
1... .... = Extension indicator: last octet
...0 0101 = User information layer 1 protocol: Recommendation H.221 and
H.242 (0x05)
1... .... = Extension indicator: last octet
Display 'rmohamed\000'
Information element: Display
Length: 9
Display information: rmohamed\000
Calling party number: '42343'
Information element: Calling party number
Length: 6
.... 1001 = Numbering plan: Private numbering (0x09)
.110 .... = Number type: Abbreviated number (0x06)
0... .... = Extension indicator: information continues through the next
octet
.... ..00 = Screening indicator: User-provided, not screened (0x00)
.00. .... = Presentation indicator: Presentation allowed (0x00)
1... .... = Extension indicator: last octet
Calling party number digits: 42343

```

SIP to H.323

When the sipd receives an INVITE with the P-Asserted-Identity header but without the Privacy header, the Net-Net SBC will set the presentation indicator to **allowed** in H.323's SETUP.

When the Privacy header is present with the value *"id"*, the presentation indicator will be set to **restricted**. The Net-Net SBC does not support any other value for the Privacy header and so for those call flows, the presentation indicator will be absent in the SETUP.

Example 1: INVITE from sip EP to sipd

```

Apr 20 04:43:54.220 On [0:0]192.168.200.68:5060 received from
192.168.200.6:5062
INVITE sip:800@192.168.200.68:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.200.6:5062
From: sipp <sip:7789@192.168.200.6:5062>;tag=1
To: sut <sip:800@192.168.200.68:5060>
P-Asserted-Identity: sipp <sip:7789@192.168.200.6:5062>
Call-ID: 1.1336.192.168.200.6@sipp.call.id
Cseq: 1 INVITE
Contact: sip:7789@192.168.200.6:5062
Max-Forwards: 70
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 136
^M
v=0
o=user1 53655765 2353687637 IN IP4 127.0.0.1
s=-
t=0 0
c=IN IP4 127.0.0.1
m=audio 10000 RTP/AVP 0
a=rtpmap:0 PCMU/8000

```

Example: INVITE from sipd to h323d

```

Apr 20 04:43:54.240 On 127.0.0.1:5070 received from 127.0.0.1:5060
INVITE sip:800@127.0.0.1:5070;acme_sag=sag1;acme_irealm=internal SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:5060;branch=z9hG4bK000c0210385hv9gpt001.1
From: sipp <sip:7789@192.168.200.6:5062>;tag=SDk0jpc01-1
To: sut <sip:800@192.168.200.68:5060>
Call-ID: SDk0jpc01-8e15e11e7f9a20523462972843c7e579-06ahc21
Cseq: 1 INVITE
Contact: <sip:7789@192.168.200.68:5060;transport=udp>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 136
GenericID: 160400004@000825010100

v=0
o=user1 53655765 2353687637 IN IP4 127.0.0.1
s=-
t=0 0
c=IN IP4 127.0.0.1
m=audio 10000 RTP/AVP 0
a=rtpmap:0 PCMU/8000

```

Sample SETUP sent from h323d to remote H323 EP

Q.931

Protocol discriminator: Q.931

Call reference value length: 2

Call reference flag: Message sent from originating side

Call reference value: 664D

Message type: SETUP (0x05)

Bearer capability

Information element: Bearer capability

Length: 3

...1 0000 = Information transfer capability: 3.1 kHz audio (0x10)

.00. = Coding standard: ITU-T standardized coding (0x00)

1... = Extension indicator: last octet

...1 0000 = Information transfer rate: 64 kbit/s (0x10)

.00. = Transfer mode: Circuit mode (0x00)

1... = Extension indicator: last octet

...0 0011 = User information layer 1 protocol: Recommendation G.711 A-law (0x03)

1... = Extension indicator: last octet

Display 'sipp'

Information element: Display

Length: 4

Display information: sipp

Calling party number: '7789'

Information element: Calling party number

Length: 6

.... 1001 = Numbering plan: Private numbering (0x09)

.110 = Number type: Abbreviated number (0x06)

0... = Extension indicator: information continues through the next octet

.... ..00 = Screening indicator: User-provided, not screened (0x00)

.00. = **Presentation indicator: Presentation all** 1... = Extension indicator: last octet

Calling party number digits: 7789

Trunk Group URIs

The Net-Net SBC's trunk group URI feature, applicable for SIP and IWF signaling services, enables the capabilities related to trunk groups that are described in this section. This implementation follows the IPTEL draft "Representing Trunk Groups in Tel/SIP Uniform Resource Identifiers (URIs)" (draft-ietf-ipitel-trunk-group-06.txt), and also supports more customized approaches.

- For a typical access call flow scenario, when the calling party's call arrives at the Net-Net SBC, the Net-Net SBC formulates a SIP INVITE message that it sends to a softswitch. The Net-Net SBC now supports a new URI contact parameter in the SIP request message so that service providers need to be able to:
 - Determine from where the Net-Net SBC received the call
 - Signal information about the originating gateway from a Net-Net SBC to a softswitch (e.g., an incoming trunk group or a SIP gateway to a Net-Net SBC)
- This feature supports the signaling of routing information to the Net-Net SBC from network routing elements like softswitches. This information tells the Net-Net SBC what egress route (or outgoing trunk groups) it should choose for terminating next hops/gateways. For this purpose, new SIP URI parameters in the Request-URI are defined. Additional URI parameters include the network context to identify the network in which the originating or terminating gateway resides.
- Especially important for large business applications, this feature can free Net-Net SBC resources by reducing the number of local policy, session agent, and session agent group configurations. By enabling the trunk group URI feature, the Net-Net instead uses a routing scheme based on signaled SIP URI information.

Terminology

The following IPTEL terms are used in the descriptions of and instructions for how to configure this feature:

- Trunk—In a network, a communication path connecting two switching systems used in the establishment of an end-to-end connection; in selected applications, it may have both its terminations in the same switching system
- Trunk group—A set of trunks, traffic engineered as a unit, for the establishment of connections within or between switching systems in which all of the paths are interchangeable except where sub-grouped
- Trunk group name—Provides a unique identifier of the trunk group; referred to as tgrp
- Trunk group context—Imposes a namespace by specifying a domain where the trunk groups are; also referred to simply as "context"

Trunk Group URI Parameters

Trunk group URI parameters identify originating and terminating trunk group information in SIP requests.

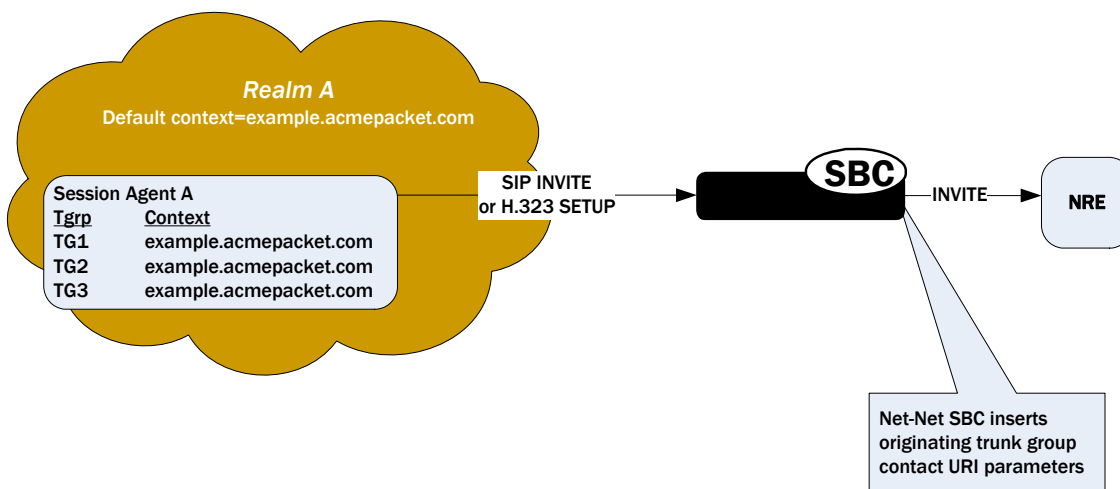
In the absence of official SIP standards for transporting trunk groups between signaling elements, the Net-Net SBC allows you to define URI parameters for use with originating and terminating trunk group URIs.

Originating Trunk Group URI Parameters and Formats

You can configure session agents and session agents groups on the Net-Net SBC to insert trunk group URI parameters in the SIP contact header. When SIP gateways comply with the IPTEL draft, they include the originating URI parameter in the SIP contact header. For those SIP and H.323 gateways that are not compliant, the Net-Net SBC inserts SIP trunk group URI parameters on the gateway's behalf.

When there are no applicable session agent or session agent group configurations, the Net-Net SBC uses the source IP address of the endpoint or gateway as the trunk group name (tgrp) parameter in the originating trunk group URI.

The following diagram shows a scenario where the Net-Net SBC inserts originating trunk group URI parameters.



There are two available formats for the originating trunk group URIs:

1. In compliance with the IPTEL draft, the first format has two parameters: tgrp (identifier of the specific trunk group) and trunk-context (defines the network domain of the trunk group). These appear in the following formats:

- tgrp="trunk group name"
- trunk-context="network domain"

The URI BNF for would appear as it does in the example directly below, where the tgrp is tg55 and the trunk-context is trunk-context is telco.example.com:

tel:+15555551212;tgrp=tg55;trunk-context=telco.example.com

2. The second format is customized specifically for access URIs and contains two provisioned parameters: tgrp (or tgrname) and context (or provstring). This appears as tgrp.context, where these definitions apply:
 - tgrp (tgrname)—Provisioned trunk group name for the originating session agent; this value must have at least one alphabetical character, cannot contain a period (.), and can contain a hyphen (-) but not as the first or the last character
 - context (provstring)—Name of the originating trunk group context; this value must have at least one alphabetical character in the top label

This format conforms to format for a hostname in the SIP URI as specified in RFC 3261, such that a trunk group identifier would appear as:

```
custsite2NY-00020.type2.voip.carrier.net
```

where the tgrp is custsite2NY-00020, and the context is type2.voip.carrier.net.

The BNF for an access URI conforms to the following:

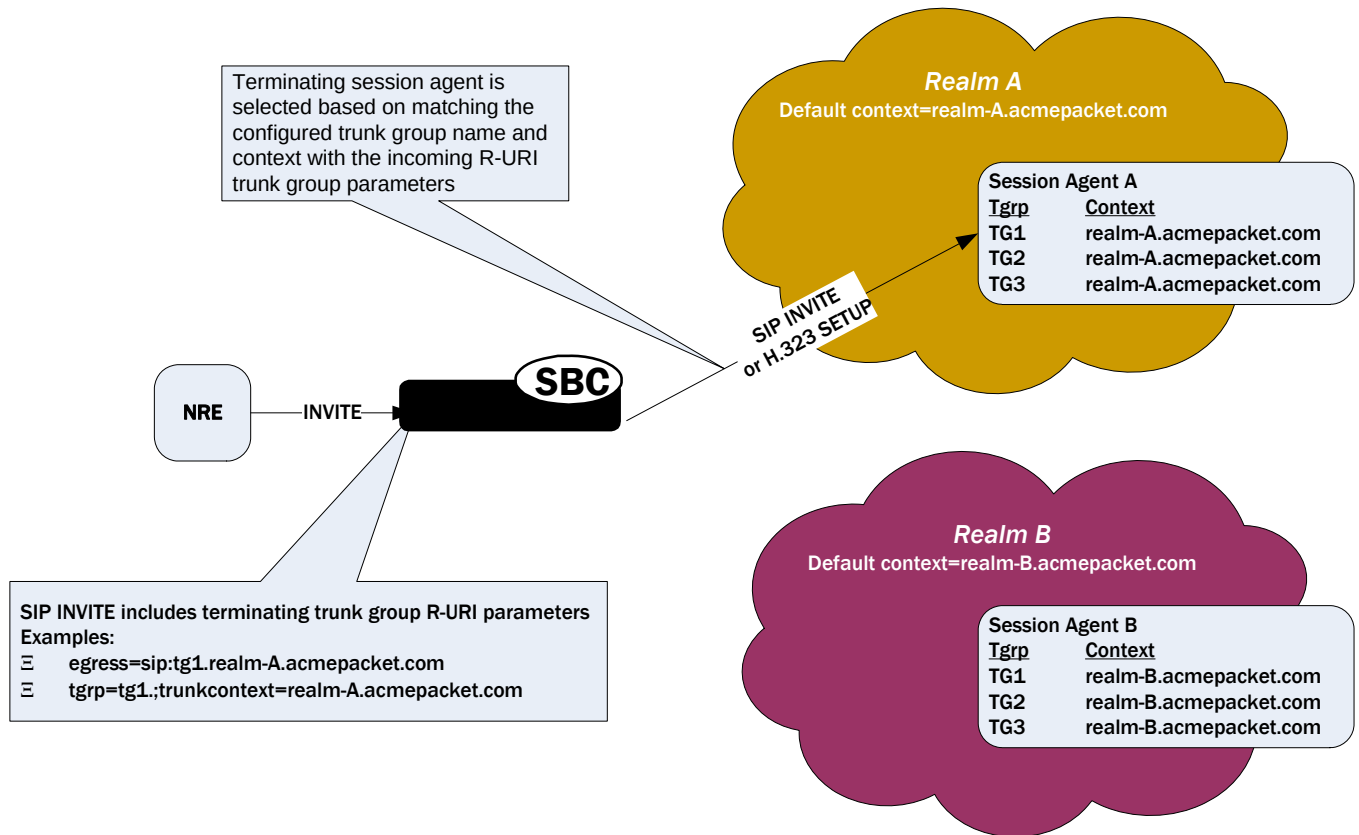
```
SIP-URI = "sip:" [userinfo ] hostport uri-parameters [headers ]
uri-parameters = *( ";" uri-parameter )
uri-parameter = transport-param / user-param / method-param
/ ttl-param / maddr-param / lr-param / other-param

other-param = accessid / pname [ '=' pvalue ]
accessid = "access=" accessURI

accessURI = scheme tname [ "." provstring ]
scheme = "sip:" / token
tname = ALPHA / *(alphanum) ALPHA *(alphanum / "-") alphanum /
alphanum *(alphanum / "-") ALPHA *(alphanum) # up to 23 characters
provstring = *(domain ".") toplabel # up to 24 characters
toplabel = ALPHA / ALPHA *( alphanum / "-") alphanum
domain = alphanum/ alphanum *( alphanum / "-") alphanum
```

Terminating Trunk Group URI Parameters and Formats

Terminating trunk group URI parameters appear in the R-URI, and they can be included in by a network routing element to instruct the Net-Net SBC which egress trunk groups to use. By matching the trunk group URI parameter with configured session agents or session agent groups, the Net-Net SBC can locate the terminating gateway. The trunk group name can also be expressed as the IP address of the terminating gateway.



In the absence of official SIP standards for transporting trunk groups between signaling elements, the Net-Net allows you to define the URI parameters used in terminating trunk groups.

There are two available formats for the terminating trunk group URIs:

1. In compliance with the IPTEL draft, the first format has two parameters: tgrp (which can be either a trunk group name or an IP address) and trunk-context (defines the network domain of the trunk group). These appear in the following formats:

- tgrp="trunk group name"
- trunk-context="network domain"

An example R-URI with terminating trunk group parameters appears as follows, where the tgrp is TG2-1 and the context is isp.example.net@egwy.isp.example.net:

```
INVITE sip:+15555551212;tgrp=TG2-1;trunk-context=isp.example.net@egwy.isp.example.net SIP/2.0
```

2. The second format is customized specifically for egress URIs and contains two provisioned parameters: tgrp (or tgname) and context (or tgdomain). This appears as tgrp.context (or tgname.tgdomain), where definitions apply:
 - tgrp (tgname)—Provisioned trunk group name for the originating session agent; this value must have at least one alphabetical character, cannot contain a period (.), and can contain a hyphen (-) but not as the first or the last character

- context (tgdomain)—Name of the terminating trunk group context; this value can be up to twenty-four characters

The use of multiple terminating trunk groups is not supported.

The BNF for a single, egress URI with trunk group information conforms to:

```

SIP-URI = "sip:" [userinfo ] hostport uri-parameters [headers ]
uri-parameters = *( ";" uri-parameter )
uri-parameter = transport-param / user-param / method-param
/ ttl-param / maddr-param / lr-param / other-param

other-param = egressid / pname [ '=' pvalue ]
egressid = "egress=" egressURI
egressURI = scheme tname [ "." tgdomain]
scheme = "sip:" / token
tname = ALPHA / *(alphanum) ALPHA *(alphanum / "-") alphanum /
alphanum *(alphanum / "-") ALPHA *(alphanum) # up to 23 characters
tgdomain = *(domain ".") toplevel # up to 24 characters
toplevel = ALPHA / ALPHA *( alphanum / "-" ) alphanum
domain = alphanum/ alphanum *( alphanum / "-" ) alphanum

```

How It Works

For all trunk group URI support, you must set the appropriate parameters in the SIP manipulations configuration and in the session agent or session agent group configurations.

In the originating trunk group URI scenario, a call arrives at the Net-Net SBC from a configured session agent or session agent group. If this session agent or session agent group has the appropriate trunk group URI parameters and inbound manipulation rules configured, the Net-Net SBC then looks to the SIP manipulations configuration and add the trunk group URI information according to those rules. Those rules tell the Net-Net SBC where and how to insert the trunk group URI information, and the Net-Net SBC forwards the call.

In the terminating trunk group scenario, a call arrives at the Net-Net SBC from, for instance, a call agent. This call contains information about what trunk group to use. If the information matches a session agent or session agent group that has outbound manipulation rules configured, the Net-Net SBC will then look up the SIP manipulations configuration and strip information according to those rules. Those rules tell the Net-Net SBC where and how to remove the information, and the Net-Net SBC forwards the call.

SIP Header and Parameter Manipulation

SIP header and parameter manipulation is its own configuration where you can set up rules for the addition, removal, and modification of a SIP header or the elements of a SIP header. For example, you can set up the configuration to add a URI parameter to the URI in a SIP header or replace an FQDN with an IP address. For trunk group URI support, this configuration tells the Net-Net SBC where and how to manipulate the SIP message to use originating (access) and terminating (egress) trunk group URI parameters.

These manipulations can be applied at the realm or at the session agent level.

To learn more about SIP header manipulation, refer to the “SIP Header and Parameter Manipulation” section of this guide’s *SIP Services* chapter.

Trunk Group Routing

You can configure SIP interfaces (using Route based on trunk group parameters) to perform routing based on the trunk group information received in SIP requests. There are three options: none, IPTEL, and egress URI.

- If you leave Route based on trunk group parameters set to none (its default), the Net-Net SBC will not look for or route based on terminating trunk group URI parameters
- When you set Route based on trunk group parameters to either **iptel** or **egress-uri** and the incoming request has the trunk group parameter of this type (IPTEL or egress URI), the Net-Net SBC will select the egress next hop by matching the “tgrp” and “trunk context” with a configured session agent or session agent group.

If the received terminating trunk group URI parameters include an IP address, the egress next hop is the IP address specified. The Net-Net SBC determines the egress realm by matching the trunk context it receives with the trunk context you configure for the realm.

- If the incoming request does not have trunk group parameters or it does not have trunk group parameters of the type that you configure, the Net-Net SBC uses provisioned procedures and/or local policy for egress call routing.

The Net-Net SBC returns errors in these cases:

- If the terminating trunk group URI parameters do not identify a local Net-Net SBC session agent or session agent group, then the Net-Net SBC returns a SIP final response of “488 Not Acceptable Here.”
- If the Net-Net SBC receives a SIP INVITE with terminating trunk group URI parameters that do not match the specified syntax, the Net-Net SBC returns a 400 final response with the reason phrase Bad Egress=Parameters.

Trunk Group URIs and SIP Registration Caching

For calls where SIP registration caching is used, you will need to set certain parameters that enable the Net-Net SBC to preserve trunk group URI parameters on the outgoing side.

- For SIP-H.323 calls requiring IWF, you set the **preserve-user-info-sa option** in the session agent configuration.

Before You Configure

Before you configure your Net-Net SBC to support trunk group URIs, you need to determine:

- How you want to manipulate SIP headers (entered in the SIP header manipulations configuration)
- For terminating trunk group routing, the trunk group mode you want to use (none, IPTEL, or egress URI); this decides routing based on trunk group information
- The trunk group name and context to use entered in a session agent or session agent group configuration
- Whether you are using originating or terminating trunk group URIs (entered in the session agent configuration)
- The trunk group context for use in a realm configuration, in case the trunk group name in the session agent or session agent group does not have a context

Configuring SIP Manipulations

For detailed instructions about how to configure SIP header manipulations, refer to Configuring Header and Parameter Manipulation in this guide's *SIP Services* chapter.

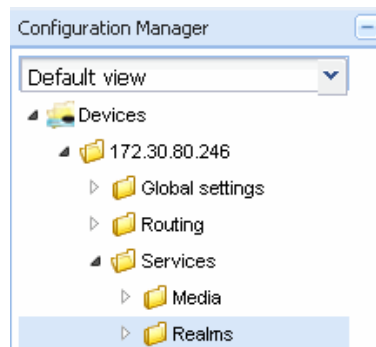
When you configure the SIP header manipulations to support trunk group URIs, take note of:

- The name of the configuration, so that you can use it when you apply the manipulations in a session agent for the inbound or outbound manipulations
- The **new-value** parameter, which specifies the trunk group and trunk group context that you want to manipulate; the possible values that apply to trunk group URI configurations are \$TRUNK_GROUP and \$TRUNK_GROUP_CONTEXT

Setting the Trunk Group URI Mode for Routing

To set the mode for routing for terminating trunk group URIs:

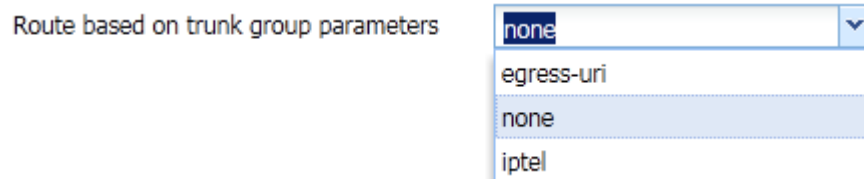
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service for which you want to configure the mode for routing for terminating trunk group URIs and click **Edit**.
6. Scroll to the Route based on trunk group parameters section of this screen. It appears just about the Charging Function area.

Route based on trunk group parameters—Choose the trunk group URI mode that you want to use from the combination box. Your choices are:

- **none**—Disables routing based on trunk groups
- **iptel**—Uses trunk group URI routing based on the IPTEL formats
- **egress-uri**—Uses trunk group URI routing based on the egress URI format



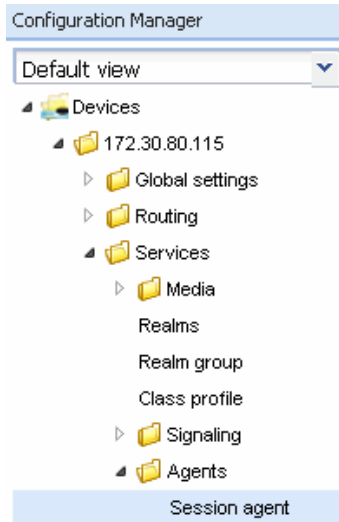
7. Click **Apply** for complete configuring the trunk group mode.

Configuring a Session Agent for Trunk Group URIs

In a session agent group, you can configure inbound or outbound SIP header manipulations rules to use, as well as a list of trunk group names. For the trunk group names, you can use either the IPTEL or the custom format.

To configure a session agent for trunk group URIs:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the session agent you want to configure in the table and click **Edit**.
4. Scroll to the Trunk group(s) table and click **Add**. The Add trunk group(s) dialog box appears.
5. **Name**—Enter the trunk group name and trunk group context in one of the following formats:

tgrp:context
tgrp.context

The screenshot shows the 'Add Trunk group(s)' dialog box. It has a title bar with a close button. Inside, there is a label '*Name' followed by a text input field containing 'tgrp1:context1'. To the right of the input field is the text '(Unique required)'. At the bottom of the dialog are two buttons: 'Apply' and 'Cancel'.

Note: If you do not include a context, the Net-Net SBC uses the trunk group context you set in the realm.

6. Click **Apply**. The new entry appears in the Trunk group(s) list.
7. Click the Translation and manipulation tab at the top of the content area.
8. **Manipulation ruleset for inbound messages**—Select the SIP header manipulation you want to apply to the traffic entering the Net-Net SBC through this session agent from the combination box.

9. **Manipulation ruleset for outbound messages**—Select the SIP header manipulation you want to apply to the traffic exiting the Net-Net SBC through this session agent from the combination box.

Manipulation ruleset

Manipulation ruleset for inbound messages ▼

Manipulation ruleset for outbound messages ▼

10. Click **Apply**.
11. If you want to configure trunk group URIs for SIP-H.323 calls that use the IWF and you are using SIP registration caching, you might need to add the **preserve-user-info-sa** to your list of session agent options. In the session agent data window, click **Options...**

The Options dialog box appears.

12. **User defined option**—Type in **preserve-user-info-sa**, making sure that the syntax is correct.

User defined option	preserve-user-info-sa	Add

13. Click **Add** to add the option. The preserve-user-info-sa window appears in the middle of the SIP options window.

Selected options

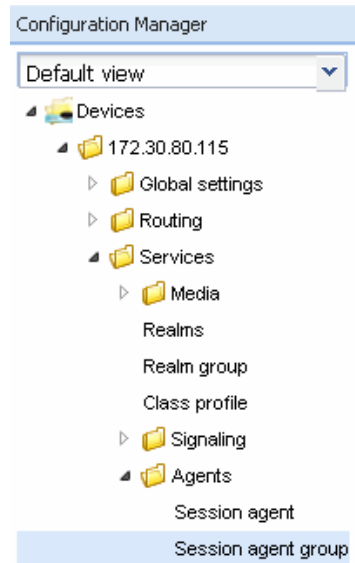
preserve-user-info-sa

14. Click **OK** to continue.
15. Click **Apply**.

Configuring a Session Agent Group for Trunk Groups URIs

You can also add a list of trunk group URIs to use for matching to a session agent group.

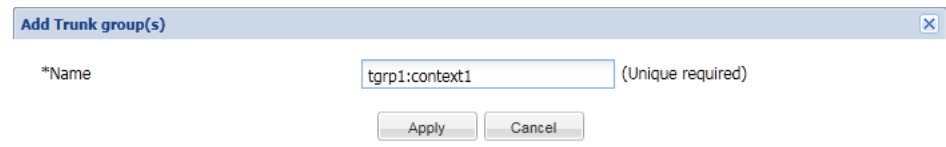
1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent group. The Session agent group table appears in the content area.
3. Select the session agent group whose trunk group URI list you want to configure and click **Edit**.
4. Scroll to the Trunk group(s) table and click **Add**. The Add trunk group(s) dialog box appears.
5. **Name**—Enter the trunk group name and trunk group context in one of the following formats:

tgrp:context

tgrp.context



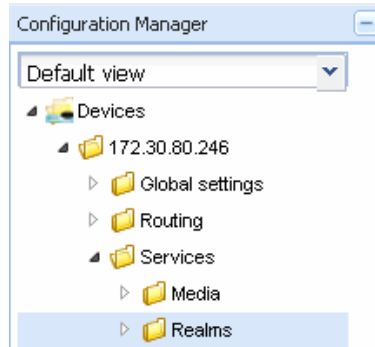
Note: If you do not include a context, the Net-Net SBC uses the trunk group context you set in the realm.

6. Click **Apply**. The new entry appears in the Trunk group(s) list.
7. Click **Apply**.

Setting a Trunk Group Context in a Realm

You can also set a trunk group context for a realm which will be used by all session agents and session agent groups if there is no context specified in their configurations.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. **Trunk context**—Enter the context you want to set.

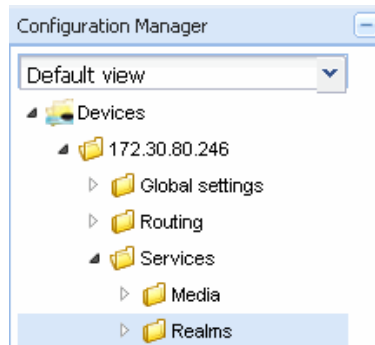
Trunk context

5. Click **Apply**.

Trunk Group URIs with SIP Interface Registration Caching

If you are using trunk group URIs with SIP interfaces that have registration caching enabled, then you need to configure the **preserve-user-info** option for that SIP interface.

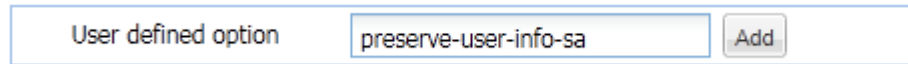
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table for which you want to configure the mode for routing for terminating trunk group URIs and click **Edit**.
6. Scroll down to the Options section of the window.
7. Click **Options...** to add new options to the SIP interface.

The SIP options window appears.

8. **User defined option**—Type in **preserve-user-info-sa**, making sure that the syntax is correct.



A screenshot of a configuration window showing a text input field labeled "User defined option" containing the text "preserve-user-info-sa". To the right of the input field is a button labeled "Add".

9. Click **Add** to add the option. The preserve-user-info-sa window appears in the middle of the SIP options window.



A screenshot of a configuration window titled "Selected options". Inside the window, there is a list box containing the text "preserve-user-info-sa".

10. Click **OK** to continue.
11. Click **Apply**.

Options for Calls that Require the IWF

You can configure several specific behaviors by configuring options for calls that require the IWF, and set them for the H.323 side of the call. These options are listed and defined in the table below. Options can be configured either globally for the H.323 configuration, individually for an H.323 interface, or for H.323 session agents.

Options	Description
MapG729	Net-Net SBC maps H.245 G.729 to SDP G.729 with Annex B and vice versa. Applicable only to calls that require the IWF.
ColonG729	Net-Net SBC uses the : (colon) instead of the = (equal sign) in the media attribute line a=fmtp:18 annexb=yes/no when mapping H.245 G.729 or SDP G.729 with Annex B. Applicable only to calls that require the IWF.
IwfLRQ	Net-Net SBC sends an INVITE (with no SDP) to a redirect server in response to an incoming LRQ received on an H.323 interface. If a 3xx message with a redirected contact header is returned, the Net-Net SBC will send an LCF in response to the LRQ. Otherwise, it will send an LRJ.
NoG729AnnexB	SDP received by the IWF with H.729 and no FMTP will be mapped to G.729 on the H.323 side of the call. Can also be set in the session agent options parameter.
sameT38Port	Net-Net SBC's H.323 process does not allocate separate ports for audio and T.38. Net-Net SBC will send the same audio port in the OLCAck that it sees in a request mode for T.38 and a new OLC for T.38.
pvtStats	Net-Net SBC includes program value tree (PVT) statistics in the show h323d display that are a sum of the PVT statistics for all H.323 interfaces. Used for debugging purposes.
acceptAI	Net-Net SBC accepts all the codecs received in the SIP 200OK and builds the TCS accordingly.

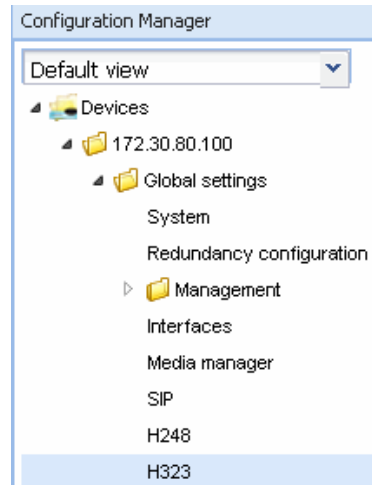
Configuring Options for Calls that Require the IWF

Follow the instructions in this section to add H.323 options for calls that require IWF using the Net-Net Central.

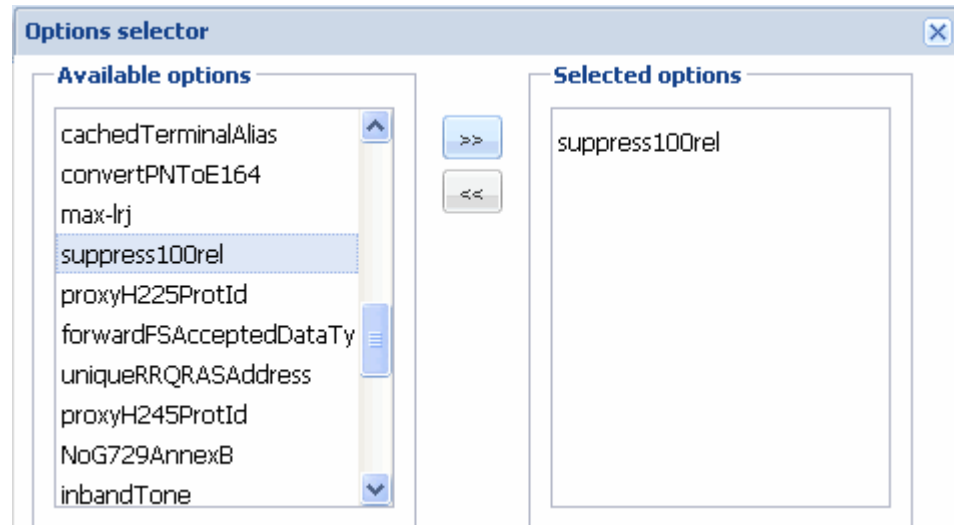
Global Options

To configure options globally for H.323:

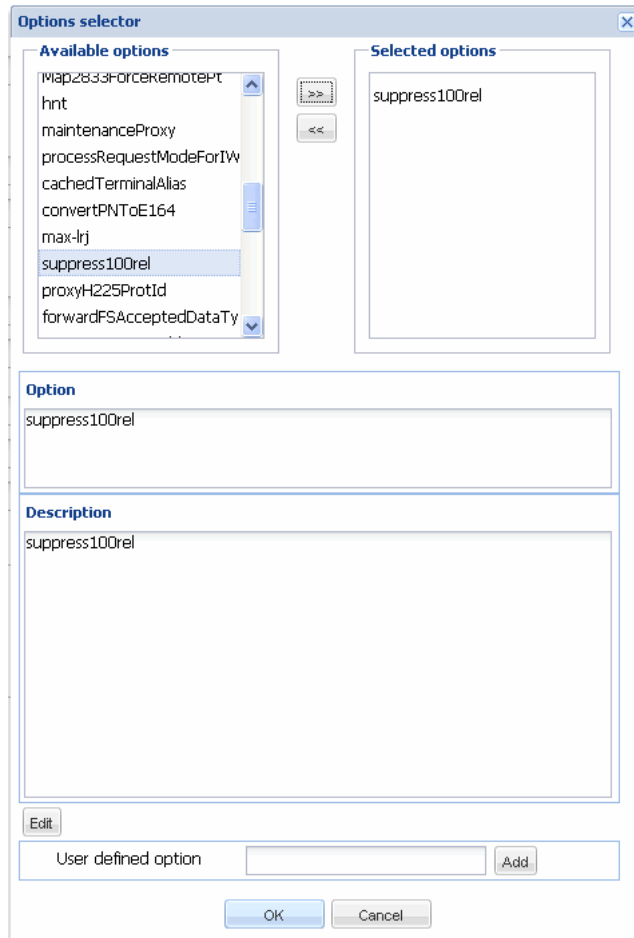
1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Scroll to Options and click **Options**.
5. Click a global option name to select it in the Select options list. For example,



6. Click  to move the option to the opposite list. The option name and details appear at the bottom of the screen



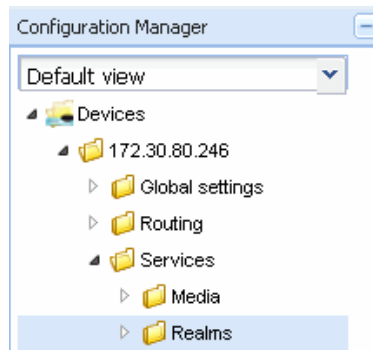
The **Options selector** dialog box is shown. It contains two lists: **Available options** and **Selected options**. The **Available options** list includes: wlap2833-forceRemotePc, hnt, maintenanceProxy, processRequestModeForIw, cachedTerminalAlias, convertPNTtoE164, max-lrj, **suppress100rel** (highlighted), proxyH225ProtId, and forwardFSAcceptedDataTy. The **Selected options** list contains: suppress100rel. Below the lists are fields for **Option** (suppress100rel) and **Description** (suppress100rel). At the bottom, there is an **Edit** button, a **User defined option** text box with an **Add** button, and **OK** and **Cancel** buttons.

7. Repeat steps 5 and 6 to add additional options.
8. Click **OK**.
9. Click **Apply**.

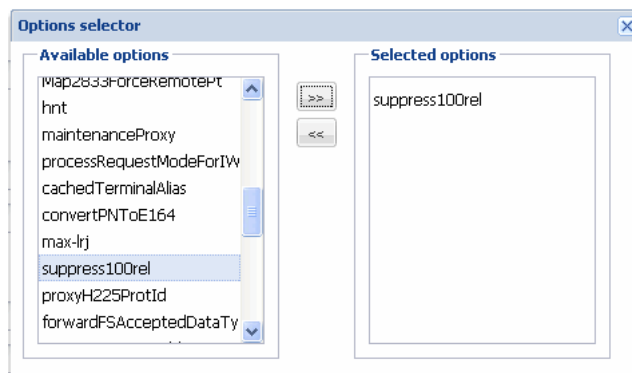
Interface Options

To configure options per individual H.323 interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 stack service in the table and click **Edit**.
6. Click the H.323 interface tab at the top of the content area.
7. Click **Options....** The H.323 option window appears.
8. Select the H.323 option that you want to use from the Available options list.

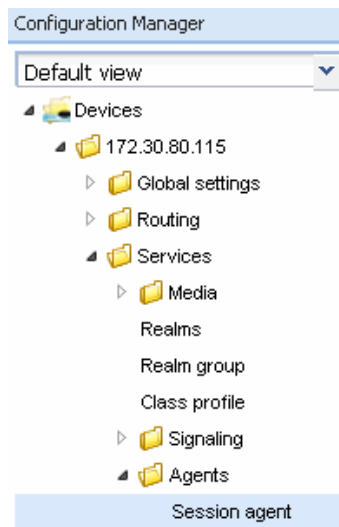


9. Click  to move the selected option to the opposite list of configured options. The option name and description appear at the bottom of the screen.
10. Click **OK**.
11. Click **Apply**.

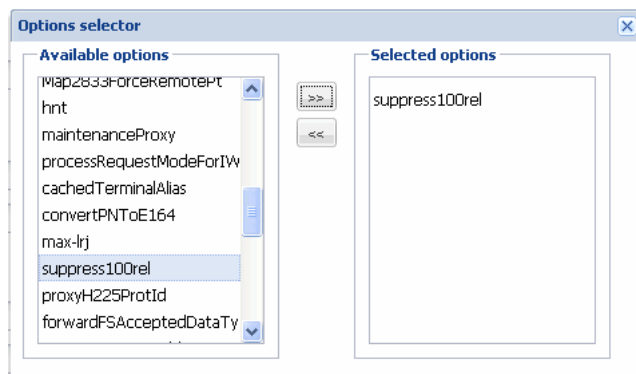
Session Agents

To configure options for H.323 session agents:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the H.323 session agent from the table to which you would like to add options.
4. Click **Edit**.
5. Click **Options**. The H.323 option dialog box appears.
6. Select the H.323 option that you want to use from the Available options list.



7. Click  to move the selected option to the opposite list of configured options. The option name and description appear at the bottom of the screen.
8. Click **OK**.
9. Click **Apply**.

IWF Privacy for Business Trunking

The Net-Net SBC's provides for two types of IWF caller privacy:

- Caller privacy on unsecure networks—Enables bi-directional SIP-H.323 IWF support for calling party identification (CPID) hiding using the presentation indicators in the Calling Party Number information element for H.323 signaling, and RFC 3325-based privacy support for SIP signaling. It lets the Net-Net SBC insert the P-Asserted-Identity and the Privacy header in the INVITE when the presentation indicator is set to *restricted*.

The presence or absence of P-Asserted-Identity and Privacy headers in the SIP INVITE informs the remote SIP proxy or endpoint either to block or advertise the CPID.

- Caller privacy on secure connections—Before the introduction of this feature, when the H.323 endpoint sent a SETUP with its presentation indicator set to *allowed*, the Net-Net SBC did not insert the P-Asserted-Identity header in the SIP INVITE. The SIP INVITE, however, needs the P-Asserted-Identity header to support calling line identification presentation (CLIP) to calling line identification restriction (CLIR) in an IP multimedia (IMS) solution. This feature lets the Net-Net SBC insert the P-Asserted-Identity in the SIP INVITE when the presentation indicator is set to *allowed*.
 - CLIP is a service provided to the called party that allows the display of the calling number (Caller ID). The user-provided calling number must be transported from the caller to the called party.
 - CLIR is a service provided to the calling party that lets it indicate whether or not the calling number is to be displayed to the called party. It sets a calling number presentation indicator to *allowed* or to *restricted*. Regulations require that network administrations remove the calling number before it is sent to the called party, if the calling party has so requested.

IWF privacy for business trunking, supports the case where SIP and H.323 PBXs are connected to the core IMS system; traffic originated at the IP PBXs terminates either at other PBXs or at the PSTN, and includes the possibility of accepting incoming traffic from the PSTN. CLIP and CLIR must be supported for calls in either direction for calls that require interworking between SIP and H.323. Unlike the two features described above, this new feature supports the fact that only a network-based application server has sufficient privilege to assert the identity of the calling party.

Thus, for this feature, the Net-Net SBC does not force privacy. Instead, the implemented feature assumes that the H.323 session agent is an IP PBX, and the Net-Net SBC only indicates to the SIP core that privacy is being requested. In other words, the Net-Net SBC is not required to interwork the H.323 presentation indicator parameter to RFC 3325 by including the P-Asserted-Identity header. The indication to the SIP core that privacy is being requested excludes identity assertion.

You configure this feature using two session agent options:

- *allowCPN*—Set in the egress H.323 session agent, allows the Net-Net SBC to send the calling party number information element (IE), even when the presentation indicator is set to *restricted*.
- *NoPAssertedId*—Set in the ingress H.323 session agent; when the incoming SETUP message has the presentation indicator is set to *restricted*, instructs the Net-Net SBC to send a Privacy header without the P-Asserted-Identity and not to make the From header anonymous.

A Call Originating in H.323

This section describes for the IWF privacy for business trunking feature works for a call originating in H.323 that requires interworking to SIP.

When the Net-Net SBC receives an H.323 SETUP with a presentation indicator of the calling party information element (IE) is set to `restricted` and this SETUP was received from a session agent is configured with the `NoPAssertedID` option, the Net-Net SBC only adds the Privacy header with the value `no`. In this case, there will be no P-Asserted-Identity and the From header will contain the calling Party information that was extracted from the callingPartyIE. The Net-Net SBC assumes that the PBX will send the callingPartyNumber in the IE, even though it would like to have the calling party number restricted.

Sample SETUP Message from an H.323 Endpoint

```
Q.931
  Protocol discriminator: Q.931
  Call reference value length: 2
  Call reference flag: Message sent from originating side
  Call reference value: 2FB6
  Message type: SETUP (0x05)
  Bearer capability
    Information element: Bearer capability
    Length: 3
    ...0 1000 = Information transfer capability: Unrestricted digital
information (0x08)
    .00. .... = Coding standard: ITU-T standardized coding (0x00)
    1... .... = Extension indicator: last octet
    ...1 0011 = Information transfer rate: 384 kbit/s (0x13)
    .00. .... = Transfer mode: Circuit mode (0x00)
    1... .... = Extension indicator: last octet
    ...0 0101 = User information layer 1 protocol: Recommendation
H.221 and H.242 (0x05)
    1... .... = Extension indicator: last octet
  Display 'jdoe\000'
    Information element: Display
  Length: 9
    Display information: jdoe\000
  Calling party number: '42343'
    Information element: Calling party number
    Length: 6
    .... 0001 = Numbering plan: E.164 ISDN/telephony numbering (0x01)
    .000 .... = Number type: Unknown (0x00)
    0... .... = Extension indicator: information continues through
the next octet
    .... ..00 = Screening indicator: User-provided, not screened
(0x00)
    .01. .... = Presentation indicator: Presentation restricted
(0x01)
    1... .... = Extension indicator: last octet
    Calling party number digits: 42343
    E.164 Calling party number digits: 42343
  Called party number: '780'
    Information element: Called party number
    Length: 4
    .... 0001 = Numbering plan: E.164 ISDN/telephony numbering (0x01)
    .000 .... = Number type: Unknown (0x00)
    1... .... = Extension indicator: last octet
```

Called party number digits: 780
 E.164 Called party number digits: 780
 User-user
 Information element: User-user
 Length: 161
 Protocol discriminator: X.208 and X.209 coded user information

Sample INVITE from the Net-Net SBC to the SIP Endpoint

```
May  5 15:11:51.996 On [0:0]192.168.200.68:5060 sent to
192.168.200.6:5060
INVITE sip:780@192.168.200.6:5060 SIP/2.0
Via: SIP/2.0/UDP
192.168.200.68:5060;branch=z9hG4bK00020a20eg11s94pg700.1
Contact: "jdoe"<sip:42343@192.168.200.68:5060;transport=udp>
Supported: 100rel
From: "jdoe"<sip:42343@192.168.200.68:5060>;tag=SDetur801-
00000194000e2ce8
To: <sip:780@192.168.200.6:5060>
Call-ID: SDetur801-231c7b30909ca525ce12cbfeb57754ea-06ahc21
CSeq: 2 INVITE
Privacy: id
Content-Length: 231
Content-Type: application/sdp
Max-Forwards: 70

v=0
o=IWF 2 2 IN IP4 192.168.200.65
s=H323 Call
c=IN IP4 192.168.200.65
t=0 0
m=audio 5004 RTP/AVP 8 0
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000/1
m=video 5006 RTP/AVP 31 34
a=rtpmap:31 H261/8000
a=rtpmap:34 H263/9000/1
```

A Call Originating in SIP

This section describes for the IWF privacy for business trunking feature works for a call originating in SIP that requires interworking to H.323.

When the Net-Net SBC receives a SIP INVITE with a Privacy header that has the value `id`, it sets the presentation indicator to `restricted` in the corresponding H.323 SETUP message. If the H.323 session agent is configured with the `allowCPN` option, the Net-Net SBC sends the display IE and the calling party number to the H.323 session agent. If that option is not set in the H.323 session agent, then the Net-Net SBC reverts to its default behavior, which is to not to send the display IE and to hide the calling party number.

Sample INVITE from a SIP Endpoint to the Net-Net SBC

```
May  5 14:41:54.513 On [0:0]192.168.200.68:5060 received from
192.168.200.6:5060
INVITE sip:800@192.168.200.68:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.200.6:5060
From: sipp <sip:sipp@192.168.200.6:5060>;tag=1
To: sut <sip:800@192.168.200.68:5060>
Call-ID: 1.3068.192.168.200.6@sipp.call.id
```

```

Cseq: 1 INVITE
Contact: sip:sipp@192.168.200.6:5060
Privacy: id
P-Asserted-Identity: sipp <sip:1234@192.168.200.6:5060>
Max-Forwards: 70
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 136

```

```

v=0
o=user1 53655765 2353687637 IN IP4 127.0.0.1
s=-
t=0 0
c=IN IP4 127.0.0.1
m=audio 10000 RTP/AVP 0
a=rtpmap:0 PCMU/8000

```

Sample SETUP from the Net-Net SBC to the H.323 Endpoint

```

Q.931
  Protocol discriminator: Q.931
  Call reference value length: 2
  Call reference flag: Message sent from originating side
  Call reference value: 44B0
  Message type: SETUP (0x05)
  Bearer capability
    Information element: Bearer capability
    Length: 3
    ...1 0000 = Information transfer capability: 3.1 kHz audio (0x10)
    .00. .... = Coding standard: ITU-T standardized coding (0x00)
    1... .... = Extension indicator: last octet
    ...1 0000 = Information transfer rate: 64 kbit/s (0x10)
    .00. .... = Transfer mode: Circuit mode (0x00)
    1... .... = Extension indicator: last octet
    ...0 0011 = User information layer 1 protocol: Recommendation
G.711 A-law (0x03)
    1... .... = Extension indicator: last octet
  Display 'sipp'
    Information element: Display
    Length: 4
    Display information: sipp
  Calling party number: '1234'
    Information element: Calling party number
  Length: 6
    .... 0001 = Numbering plan: E.164 ISDN/telephony numbering (0x01)
    .010 .... = Number type: National number (0x02)
    0... .... = Extension indicator: information continues through
the next octet
    .... ..00 = Screening indicator: User-provided, not screened
(0x00)
    .01. .... = Presentation indicator: Presentation restricted
(0x01)
    1... .... = Extension indicator: last octet
    Calling party number digits: 1234
    E.164 Calling party number digits: 1234
  Called party number: '800'
    Information element: Called party number

```

```

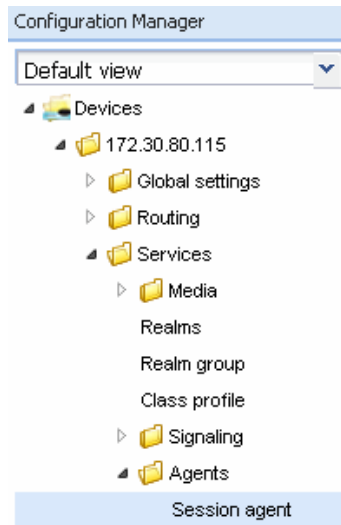
Length: 4
.... 0001 = Numbering plan: E.164 ISDN/telephony numbering (0x01)
.010 .... = Number type: National number (0x02)
1... .... = Extension indicator: last octet
Called party number digits: 800
E.164 Called party number digits: 800
User-user
Information element: User-user
Length: 159
Protocol discriminator: X.208 and X.209 coded user information

```

Net-Net Central Instructions and Examples

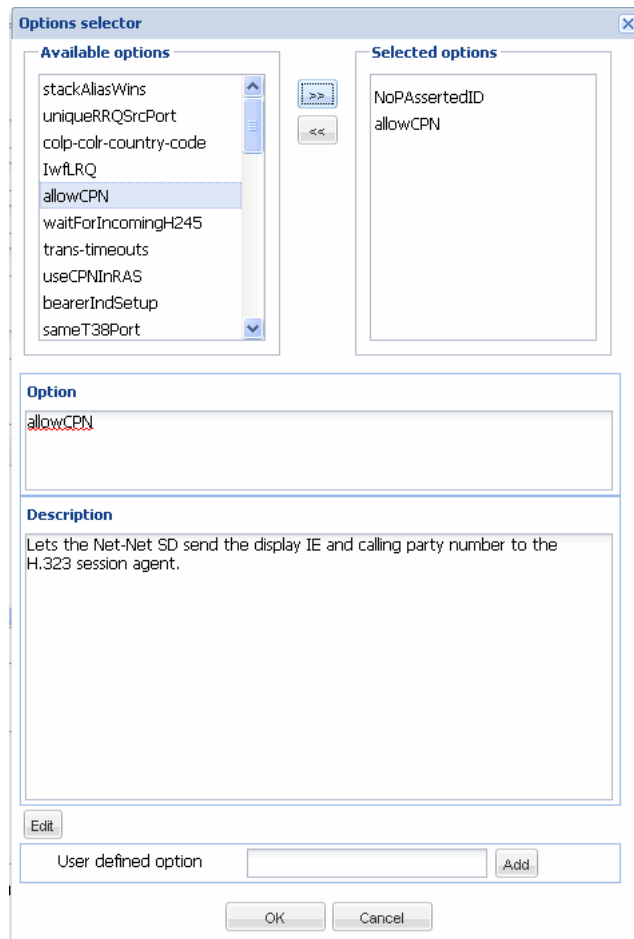
You can set both of these options in the same H.323 session agent.

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the H.323 session agent from the table to which you would like to add options.
4. Click **Edit**.
5. Click **Options**. The H.323 option dialog box appears.
6. Click **allowCPN** in the Available options list to select it.
7. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.

8. Repeat steps 5 and 6 to choose the `NoPAssertedID` option.



9. Click **OK**.
10. Click **Apply**.

IWF COLP/COLR Support

When you enable the connected line identity presentation (COLP) and connected line identity restriction (COLR) feature for calls being translated between SIP and H.323 (using the Net-Net SBC's IWF), the Net-Net SBC converts the H.323 Connected Number Information element (IE) to the SIP P-Asserted-Identity (PAI) header and vice versa.

When there is no Q.931 Connected Number IE, the Net-Net SBC converts the H.225 Connected Address alias (either E.164 or Public Party Number).

How It Works

This section describes show the IWF COLP/COLR feature works for IWF calls that originate in SIP and are translated to H.323, and for calls that originate in H.323 and are translated to SIP.

SIP to H.323 Calls

For this type of call, the Net-Net SBC checks the Connect that it receives for a Q.931 Connected Number IE. If it does not find one, then it continues by checking for

H.225 Connected Address alias (either E.164 or Public Party Number). Then, it takes one of the following courses of action depending on circumstances:

- If it finds the Q.931 Connected Number IE, the Net-Net SBC extracts the screening indicator and the presentation indicator.
- If there is no Q.931 Connected Number IE, the Net-Net SBC extracts the screening indicator and the presentation indicator from the H.225 Connect-UIE of the Connect message.

With these pieces of information in place, the Net-Net SBC performs the conversion from H.323 Connected Number IE to SIP P-Asserted-Identity (PAI) header if and only if the screening indicator is either one of the following:

- Network provided
- User-provided, verified and passed

Then the Net-Net SBC adds a SIP PAI header (with URI value) to the 200 OK message that it sends in the SIP call leg. The user part of the URI is set to the value of the Q.931 Connected Number IE's numberDigits field, or to dialDigits value from the Connected Address alias. When the number type is a national number, the Net-Net SBC adds a plus sign (+) and the IWF country code (that you configure) to the beginning of the user part. If the number type is an international number, the Net-Net SBC only adds a plus sign (+). And when the Connected Number is empty, the Net-Net SBC sets the user part of the PAI header URI to *anonymous*. When the value in the presentation indicator is Presentation restricted, the Net-Net SBC adds the SIP Privacy header (with the value *id*) to the 200 OK.

In cases when it does not find a screening indicator, the Net-Net SBC will not perform the conversion from the H.323 Connected Number IE to the SIP P-Asserted-Identity (PAI) header.

H.323 to SIP Calls

For this type of call, the Net-Net SBC checks the 200 OK message for a SIP PAI header and a SIP Privacy header. Before it sends a Connect message on the H.323 call leg, the Net-Net SBC generates a Connected Number. It uses the Connected Number to insert a Q.931 Connected Number IE and an H.225 Connected Address alias (type E.164) into the Connect message. The Connected Number is generated in this way:

- If the
 - SIP PAI header is not found, or
 - User part of its URI value is *unknown* or *anonymous*, or
 - User part of its URI does not follow the H.225 NumberDigits syntax,

then the Connect Number that the Net-Net SBC generates is a Q.931 Connected Number IE that has no digits and a number type of *unknown*. In this case, the Net-Net SBC will not insert an H.225 Connected Address alias into the Connect message.

The presentation indicator is set to *Number not available due to interworking*, and the screening indicator to *Network provided*. The H.225 NumberDigits's syntax requires that it be between 1 and 128 characters, and only contain these characters: 0 through 9, the pound sign (#), the asterisk (*), and the comma (,).

- In all other cases, the Net-Net SBC uses the user part of the URI as the digits for the Connected Number after it performs the following:
 - Strips the plus sign in front of the number, if there is one

- Strips the IWF country code at the beginning of the number, if there is one

Then the Net-Net SBC inserts the Connected Number into the Connect message as the Q.931 Connected Number IE and an H.225 Connected Address alias (type E.164).

If the IWF country code is found in the PAL, the Net-Net SBC sets the type of Q.931 Connected Number IE to National Number. Otherwise, the Net-Net SBC sets it to international. The screening indicator is set to **Network provided**, and the presentation indicator is set to **Presentation Restricted** if the Net-Net SBC finds a SIP Privacy header with a value of **id**, or **Presentation Allowed** if there is not SIP Privacy header.

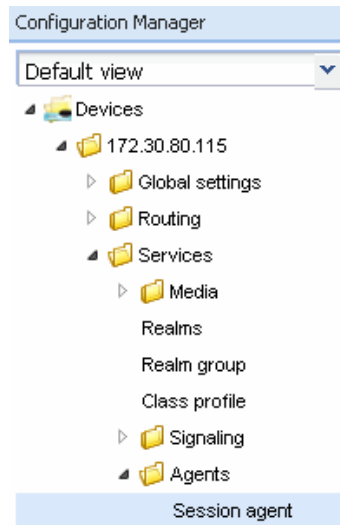
Net-Net Central Instructions and Examples

You configure IWF COLP/COLR support by setting two options:

- **colp-colr-iwf**—Setting this option enables support for IWF COLP/COLR
- **colp-colr-country-code**—Must be set if you configure the **colp-colr-iwf** option to recognize or build a national number; the value you enter here:
 - Must be a string of digits from 0 to 9
 - Cannot exceed 32 digits
 - Cannot contain any non-numeric characters; while it allows you to enter them, the system ignores any non-digits characters and so the feature might not work as needed

To enable IWF COLP/COLR support:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the H.323 session agent from the table to which you would like to add options.
4. Click **Edit**.
5. Click **Options**. The H.323 option dialog box appears.
6. Click **colp-colr-country-code** in the Available options list to select it.

7. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
 - **Description**—Edit the textbox to set the country code. Must be set if you configure the **colp-colr-iwf** option to recognize or build a national number. The value you enter here:
 - Must be a string of digits from 0 to 9
 - Cannot exceed 32 digits
 - Cannot contain any non-numeric characters; while it allows you to enter them, the system ignores any non-digits characters and so the feature might not work as needed
8. Repeat steps 5 and 6 to select to move the **colp-colr-iwf** option to the Selected options list.
9. Click **OK**.
10. Click **Apply**.

IWF Privacy Extensions for Asserted Identity in Untrusted Networks

For IWF privacy, the Net-Net SBC supports:

- IWF caller privacy on unsecure networks—A variant of RFC 3325, where the P-Asserted-Id is inserted when the presentation indicator is allowed. This feature enables bi-directional SIP/H.323 IWF support for CPID hiding by using the presentation indicators in the Calling Party Number information element for H.323 signaling, and RFC 3325-based privacy support for SIP signaling. It lets the Net-Net SBC insert the P-Asserted-Identity and the Privacy header in the INVITE when the presentation indicator is set to restricted.

The presence, or absence, of P-Asserted-Identity and Privacy headers in the SIP INVITE informs the remote SIP proxy or endpoint to either block or advertise the CPID.

- IWF caller privacy on secure connections—When the H.323 endpoint sends a SETUP with presentation indicator set to allowed, the Net-Net SBC does not insert the P-Asserted-Identity in the INVITE. The SIP INVITE needs the P-Asserted-Identity header to support calling line identification presentation (CLIP) to calling line identification restriction (CLIR) in an IP multimedia subsystem (IMS) solution. This feature lets the Net-Net SBC insert the P-Asserted-Identity in the INVITE when the presentation indicator is set to allowed.

Now the Net-Net SBC supports an enhancement to IWF caller privacy where the P-Preferred-Identity is inserted instead of the P-Asserted-Identity.

In this implementation, when the incoming H.323 Setup message has a presentationIndicator set to “restricted” and the ingress H.323 session agent has the new PPreferredId option configured, the Net-Net SBC sends the Privacy header with P-Preferred-Identity (instead of P-Asserted-Identity).

IWF Call Originating in H.323

This section shows the H.323 Setup that arrives from and H.323 endpoint, and how the Net-Net SBC adds the P-Preferred-Identity header (which has calling party number information) and the Privacy header to the SIP INVITE.

Sample H.323 Setup from a Remote Endpoint

```

Q.931
  Protocol discriminator: Q.931
  Call reference value length: 2
  Call reference flag: Message sent from originating side
  Call reference value: 2FB6
  Message type: SETUP (0x05)
  Bearer capability
    Information element: Bearer capability
    Length: 3
    ...0 1000 = Information transfer capability: Unrestricted digital information
(0x08)
    .00. .... = Coding standard: ITU-T standardized coding (0x00)
    1... .... = Extension indicator: last octet
    ...1 0011 = Information transfer rate: 384 kbit/s (0x13)
    .00. .... = Transfer mode: Circuit mode (0x00)
    1... .... = Extension indicator: last octet
    ...0 0101 = User information layer 1 protocol: Recommendation H.221 and H.242 (0x05)
    1... .... = Extension indicator: last octet
  Display 'rdoe\000'
    Information element: Display
    Length: 9
    Display information: rmohamed\000
  Calling party number: '42343'
    Information element: Calling party number
    Length: 6
    .... 0001 = Numbering plan: E.164 ISDN/telephony numbering (0x01)
    .000 .... = Number type: Unknown (0x00)
    0... .... = Extension indicator: information continues through the next octet
    .... .00 = Screening indicator: User-provided, not screened (0x00)
    .01. .... = Presentation indicator: Presentation restricted (0x01)
    1... .... = Extension indicator: last octet
    Calling party number digits: 42343
    E.164 Calling party number digits: 42343
  Called party number: '780'
    Information element: Called party number
    Length: 4
    .... 0001 = Numbering plan: E.164 ISDN/telephony numbering (0x01)
    .000 .... = Number type: Unknown (0x00)
    1... .... = Extension indicator: last octet
    Called party number digits: 780
    E.164 Called party number digits: 780
  User-user
    Information element: User-user
    Length: 161
    Protocol discriminator: X.208 and X.209 coded user information

```

Sample SIP INVITE from the Net-Net SBC to a SIP Endpoint

```

Aug 29 15:46:25.214 On [0:0]192.168.200.68:5060 sent to
192.168.200.6:5060
INVITE sip:780@192.168.200.6:5060 SIP/2.0
Via: SIP/2.0/UDP
192.168.200.68:5060;branch=z9hG4bK6810pr20205h2akqe381.1
Contact: "Anonymous"<sip:anonymous@192.168.200.68:5060;transport=udp>
Supported: 100rel
From: "Anonymous"<sip:anonymous@anonymous.invalid>;tag=SDfd9sa01-
000000ba00023280
To: <sip:780@192.168.200.6:5060>
Call-ID: SDfd9sa01-6f93292521b83a0980647f34451c5afd-06ahc21
CSeq: 2 INVITE
P-Preferred-Identity: "rdoe"<sip:42343@192.168.200.68:5060>
Privacy: id
Content-Length: 180
Content-Type: application/sdp

```

Max-Forwards: 70

v=0
o=IWF 5 5 IN IP4 192.168.200.5
s=H323 Call
c=IN IP4 192.168.200.65
t=0 0
m=audio 5010 RTP/AVP 0
a=rtpmap:0 PCMU/8000/1
m=video 5014 RTP/AVP 31
a=rtpmap:31 H261/9000/1

Before You Configure

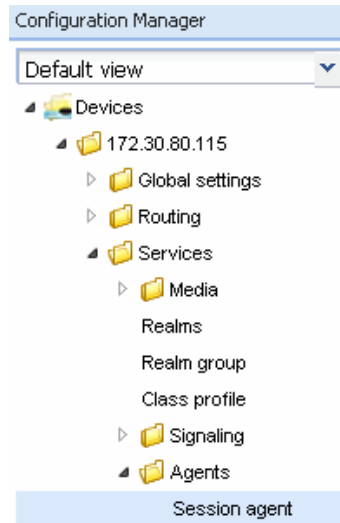
Before you configure your Net-Net SBC to support this feature, note the following considerations:

- The ingress H.323 session agent cannot be configured with the `NoPAssertedId` option
- For use in Release 4.1.1 and higher, the global SIP configuration should be configured with the `disable-ppi-to-pai` option; the older `disable-privacy` option will also work

Net-Net Central Instructions and Examples

To enable the inclusion of P-Preferred-Identity:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the H.323 session agent from the table to which you would like to add options.
4. Click **Edit**.
5. Click **Options**. The H.323 option dialog box appears.
6. Click **ppreferredid** in the Available options list to select it.
7. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
8. Click **OK**.
9. Click **Apply**.

IWF RAS Registration Failure Code Mapping

For calls that require interworking between H.323 and SIP, the Net-Net SBC supports IWF response code mapping. This feature enables the Net-Net SBC to support configurable SIP response codes for IWF calls that fail during RAS, when the Net-Net SBC has been unable to register with a gatekeeper; this allows a wider range of more accurate response codes to be communicated.

When this feature is not enabled, the Net-Net SBC generates a 404 Not Found when a SIP-to-H.323 call fails as a result of the stack's failure to register with a gatekeeper.

Supported Status Codes

When the condition noted above takes place, the response code can be any of the ones listed in this table. The code values listed in the table are used to specify the code to which you want to map.

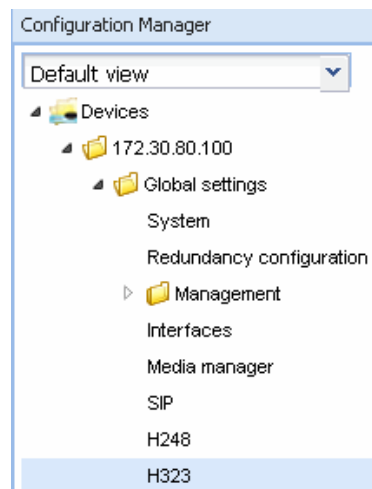
Code	Description
403	Forbidden
406	Not Acceptable
408	Request Timeout
410	Gone
420	Bad Extension
480	Temporarily Unavailable
486	Busy Here
487	Request Terminated
500	Server Internal Error
503	Service Unavailable
504	Server Time-out
600	Busy Everywhere
603	Decline

Net-Net Central Instructions and Examples

Using this feature requires that you set a new options parameter in the global H.323 configuration.

To enable IWF response code mapping:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Scroll to Options and click **Options**.
5. Click **Options**. The Options dialog box appears.
6. Click **iwfRegFailCode** in the Available options list to select it.
7. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
8. Click **OK**.
9. Click **Apply**.

IWF Codec Negotiation: H.323 Slow Start to SIP

You can enable a setting in the IWF configuration that prevents the Net-Net SBC from sending an SDP offer in the SIP INVITE when the Net-Net SBC is translating a call initiated in H.323 slow start to SIP. Instead, the Net-Net SBC expects to see an SDP offer from the SIP endpoint in a provisional, reliable provisional, or 200 OK response. Then it sends an answer in the respective ACK, PRACK, or ACK.

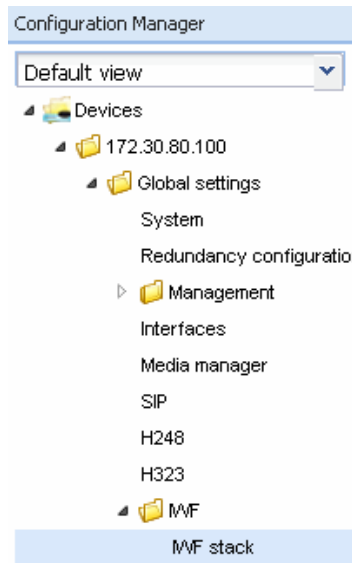
With this parameter disabled (default), as in prior Net-Net SBC releases, the Net-Net SBC populates the SIP INVITE with SDP based on the media profiles applied to the ingress H.323 session agent or the IWF configuration.

Note that the Net-Net SBC supports SIP INVITE without an SDP offer for SIP-H.323 calls, and you do not have to enable the new parameter noted below to enable this support. The Net-Net SBC sends the SDP offer in the provisional, reliable provisional, or 200 OK response. To do so, the egress H.323 session agent or the H.323 interface (stack) should be configured with media profiles.

Configuring Codec Negotiation

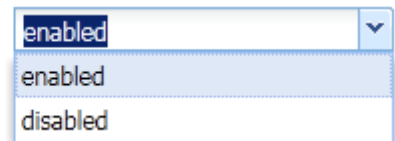
To configure codec negotiation:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click IWF stack. The IWF data window appears in the content area.
3. **Suppress SDP in the INVITE for slow-start to SIP calls**—Select enabled from the combination box to prevent the Net-Net SBC from sending an SDP offer in the SIP INVITE for an IWF call initiated in H.323 slow start (being translated to SIP).

Suppress SDP in the INVITE for slow-start to SIP calls



4. Click Apply.

IWF: H.245 Signaling Support for G.726

The Net-Net SBC supports the G.726 payload type for H.245 and calls that require interworking (IWF) between SIP and H.323.

For H.245, this feature extends the H.245 GenericCapability provided in Release 4.1.4 (available in sd414p6).

How It Works: IWF

For IWF calls using ITU-T G.726 as the audio codec, the SIP call leg requires G.726 in the SDP. The H.323 side of the call signals G.726 (in the H.245 openLogicalChannel and TerminalCapabilitySet messages) by including a GenericCapability defining G.726 as the codec. In the GenericCapability, the capabilityIdentifier and maxBitRate parameters identify G.726. While a capabilityIdentifier with 0.0.7.726.1.0 designates G.726, the maxBitRate designates the data transmission rate.

Codec	Max Bit Rate	Data Rate
G726-16	160	16 kbit/s
G726-24	240	24 kbit/s
G726-32	320	32 kbit/s
G726-40	400	40 kbit/s

To support G.726 for IWF calls, the Net-Net SBC converts the G726-X value in the SDP of SIP messages to a GenericCapability structure in H.323/H.245 messages, and the conversion works the same way in reverse.

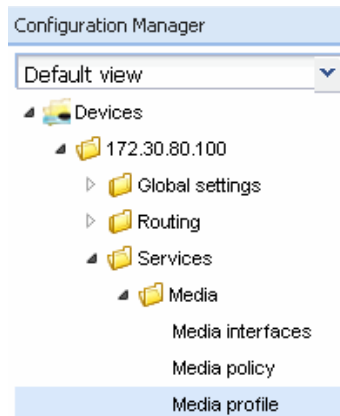
Configuring Signaling Support for G.726

Media profiles now allow you to set the configuration to any of the four G.726 encodings. (as defined by ITU G726 Annex B and RFC 3551) and to genericAudioCapability.

Creating a Generic Audio Media Profile

To set a media profile for generic audio support:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**. The Add Media profile dialog box appears.

4. **Media profile name**—Choose **genericAudio** from the combination box to set the name of the generic video media profile.

5. **Subname**—Enter a subname to distinguish this genericAudio media profile.
6. Click **Apply**. The genericAudio media profile appears in the Media profile table.
7. Click the genericAudio profile in the Media profile table to choose it and click **Edit**.
8. **Media type**—Choose **video** from the combination box as the media type to use for this media profile.

9. **Payload type**—Enter the payload type to use for the generic video media profile.

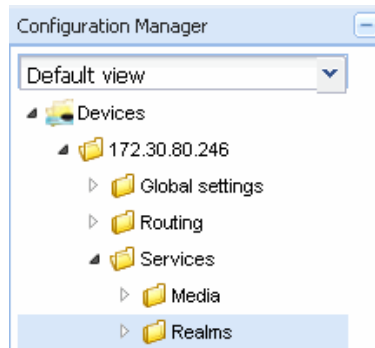
10. **Transport protocol**—Select the transport type to use for the generic video media profile from the combination box.

11. Complete the rest of the media profile configuration as needed.
12. Click **Apply**.

Configuring the H.323 Interface

To configure the H.323 interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the H.323 realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select an H.323 stack service in the table and click **Edit**.
6. Click the H.323 interworking tab at the top of the content area.
7. **Media profiles for outgoing fast start call**—Click the **Add** button in the Media Profiles section to add a media profile for an outgoing fast start call.

Media profiles for outgoing fast start call

The window titled 'Media profiles for outgoing fast start call' contains a table with one column labeled 'Name'. The table body is empty. Below the table are two buttons: 'Add' and 'Delete'.

The Add media profiles for outgoing fast start call window appears.

8. **Media profile**—Select the media profile you want to use when translating Slow Start calls to Fast Start from the combination box. This information is used to open logical channels for the outgoing call.

The window titled 'Add Media profiles for outgoing fast start call' shows a dropdown menu for '*Media profile'. The menu is open, displaying a list of options: 'telephone-event', 'G722', 'G723', 'G726-16', 'G729A', and 'G729'. The 'telephone-event' option is currently selected. To the right of the dropdown, the text '(Unique required)' is visible.

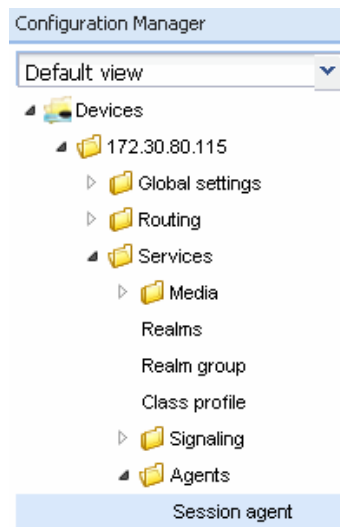
9. If you enter the name of a media profile that does not exist, the Net-Net SBC will not perform translation. If you leave this parameter empty, the Net-Net SBC will not perform translation.

10. Click **Apply**. The media profile name appears in the Media profiles for outgoing fast call list.
11. Click **Apply**.

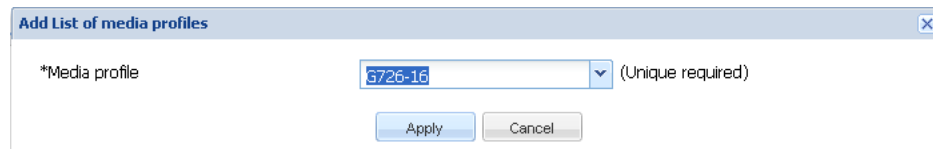
Configuring H.323 Session Agents

To configure H.323 session agents:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the H.323 session agent from the table and click **Edit**.
4. **List of media profiles**—Click **Add**.
5. **Add list of media profiles**—Choose the name of the media profile from the combination box. Values are: G726-16, G726-24, G726-32, G726-40, and genericAudioCapability.

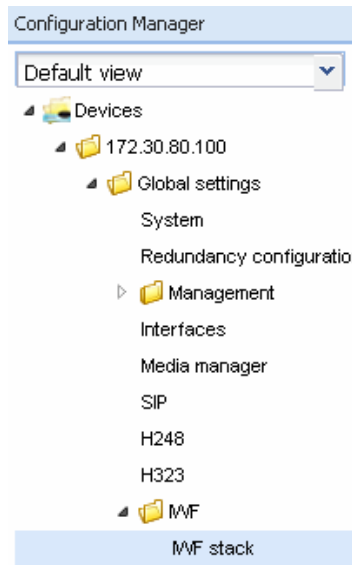


6. Click **Apply**. The media profile name appears in the table.
7. Click **Apply**.

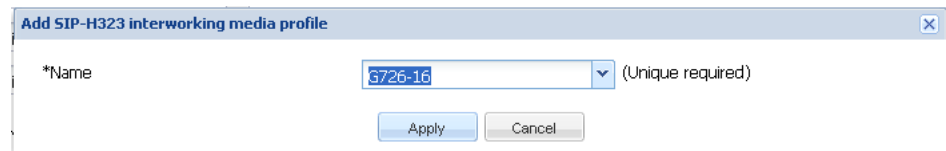
Configuring Interworking

To configure interworking:

1. Expand the Configuration Manager slider > Global settings > IWF.



2. Click IWF stack. The IWF data window appears in the content area.
3. **Media profiles for interworking incoming H323 slow start call to outgoing SIP call**—Click Add. The Add SIP-H323 interworking media profile dialog box appears.
4. **Media profile name**—Choose the name of the media profile from the combination box. Values are: G726-16, G726-24, G726-32, G726-40, and genericAudioCapability.

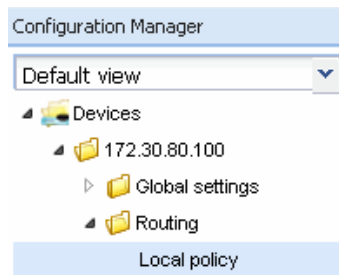


5. Click OK. The media profile name appears in the Media profiles for interworking incoming H323 slow start call to outgoing SIP call list.
6. Click **Apply**.

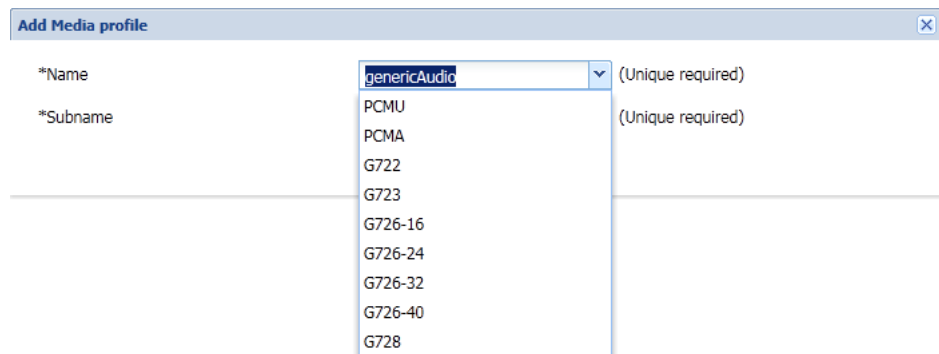
Configuring Local Policy

To configure local policy:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Select a route in the Routes table to choose it and click **Edit**.
4. Scroll to the Route policies attributes table.
5. Select a route policy in the table and click **Edit**. The Edit Route policy window appears.
6. **Media profiles**—Click **Add**. The Add profiles dialog box appears.
7. **Media Profiles**—Choose the name of the media profile from the combination box. Values are: G726-16, G726-24, G726-32, G726-40, and genericAudio.

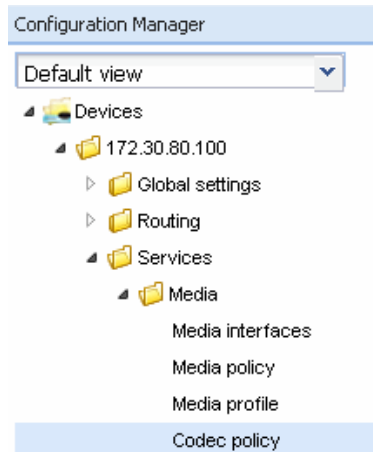


8. Click **OK**. The media profile name appears in the Media profiles list.
9. Click **Apply**.

Configuring Codec Policy

To configure codec policy:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Codec policy. The Codec policy table opens in the content area.
3. Select a Codec policy and click **Edit**.
4. Click **Add**. The Add Codec policy dialog box appears.
5. **Codec policy name**—Enter the unique name for the codec policy. You refer to this codec policy name to apply it to realms or session agents. This parameter is required.

The 'Add Codec policy' dialog box has a title bar with a close button. It contains a text input field labeled '*Codec policy name' with the value 'CodecPolicy1' and a '(Unique required)' label. Below the input field are 'Apply' and 'Cancel' buttons.

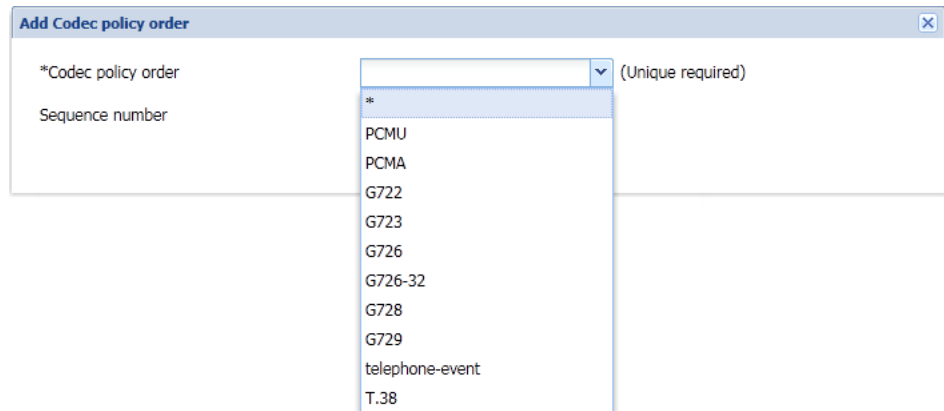
6. Click **Apply** to save the name and close the dialog box. The Codec policy name appears in the table.
7. Click the Codec in the table to select it and click **Edit**. The Edit Codec policy dialog box appears.
8. For Codec policy allow, click **Add**. The Add Codec policy allow dialog box appears.
9. **Codec policy allow**—Choose the media format type (codec) to allow for this codec policy from the combination box.

The 'Add Codec policy allow' dialog box has a title bar with a close button. It contains a dropdown menu labeled '*Codec policy allow' with the value 'PCMU' and a '(Unique required)' label. The dropdown menu is open, showing a list of options: *, PCMU, PCMA, G722, G723, G726, G726-32, G728, G729, telephone-event, and T.38. The 'G723' option is highlighted.

You can use the asterisk (*) as a wildcard, the force attribute, or the no attribute so that the allow list you enter directly reflects your configuration needs. For more information, refer to the Manipulation Modes section.

Note: The codecs that you enter here must have corresponding media profile configurations.

10. Click **Apply** to close the dialog box. The codec appears in the Codec policy allow list.
11. For Codec policy order, click **Add**. The Add Codec policy order dialog box appears.
12. **Codec policy order**—Choose the codec from the combination box.



The order in which you list the codecs is the order in which they appear in the outgoing SDP offer. You can use the asterisk (*) as a wildcard in different positions of the order to directly reflect your configuration needs. For more information, refer to the Manipulation Modes section.

Note: The codecs that you enter here must have corresponding media profile configurations.

13. Click **Apply** to close the dialog box. The codec appears in the Codec policy order list.
14. Click **Apply**.

Customized G.729 Support

The Net-Net SBC supports the use of custom G.729 encoding for calls that require interworking between SIP and H.323. If you use the proprietary G.729 encoding format in your network—G.729 that conforms neither to RFC 3551 or to RFC 3555, you might need to use this feature.

When you set the **acceptG729abFormat** option in the global H.323 configuration, the Net-Net SBC performs conversion like those in the following examples:

- For calls initiated in SIP, the Net-Net SBC can parse RTP map strings such as G.729a and G.729ab in the SDP, and then map them to H.245 data types.
 - G.729a becomes g729AnnexA.
 - G.729ab becomes g729AnnesAwAnnexB.
- For calls initiated in H.323, the Net-Net SBC can create non-standard RTP map strings such as G.729a and G.729ab from mapped H.245 data types.
 - g729 becomes G729.
 - g729AnnexA becomes G.729a.
 - g729AnnesAwAnnexB becomes G.729ab.

How It Works

When you enable the **acceptG729abFormat** option, the Net-Net SBC performs customized G.729 mapping in these instances:

- Calls initiated in SIP and translated to H.323, when the IWF:
 - Converts the SDP in an incoming SIP INVITE to a list of fastStart OpenLogicalChannel requests that are in turn included in the outgoing Setup message.
 - Converts the list of fastStart OpenLogicalChannelAck responses (which can be received in any message up to and including the Connect message) to SDP sent with a SIP response.
- Calls initiated in H.323 and translated to SIP, when the IWF:
 - Converts the list of fastStart OpenLogicalChannel requests to SDP sent with the outgoing SIP INVITE.
 - Converts SDP in a SIP response (such as a 200 OK) to the list of the list of fastStart OpenLogicalChannelAck responses included with the callProceeding, Progress, Alerting, or Connect message. This depends on when the SDP is received on the SIP side.
- For all IWF calls regardless of initiation protocol, when the IWF:
 - Converts SDP on the SIP side to the terminalCapabilitySet message to be sent on that H.323 side.

When the format is G729, the Net-Net SBC maps it to:

- g729wAnnexB if the a=fmtp:18 annexb=yes attribute is present
- g729 if the a=fmtp:18 annexb=no attribute is present
- g729wAnnexB if the no a=fmtp:18 annexb=no attribute is present

The Net-Net SBC also maps G729 to g729 because pure G729 with static payload type 18 does not include an fmtp attribute where annexb=no.

About Dynamic Payload Mapping

G.729a and G.729ab use dynamic payload types. The Net-Net SBC does not propagate these dynamic payload types to corresponding dynamicRTPPayloadType (an optional field in OpenLogicalChannel requests) on the H.323 side.

For an IWF call initiated in H.323, the dynamic payload types for G.729a and G.729ab are retrieved from media profile configurations when the Net-Net SBC converts the list of fastStart OpenLogicalChannel requests to SDP sent on the SIP side. As a result, you must set up media profile configurations for G.729a and G.729ab for the feature to work properly. In these media profiles, the following parameters must be set as follows:

- **name**—For the G.729a profile, set the **name** to **G.729a**. For the **G.729ab** profile, set the **name** to **G.729ab**.
- **payload type**—For each media profile (**G.729a** and **G.729ab**), DO NOT use payload type 18, which is the static payload type used for G.729a and G.729ab.

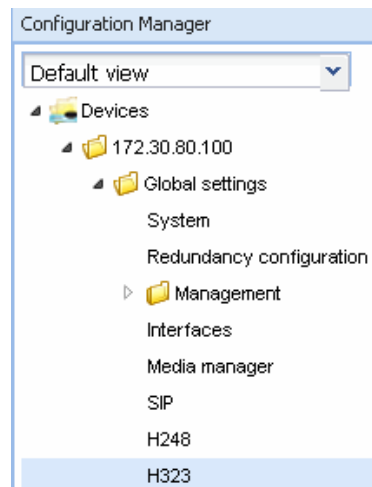
Configuring Customized G.729 Support

This section show you how to configure the **acceptG729abFormat** option in the global H.323 configuration.

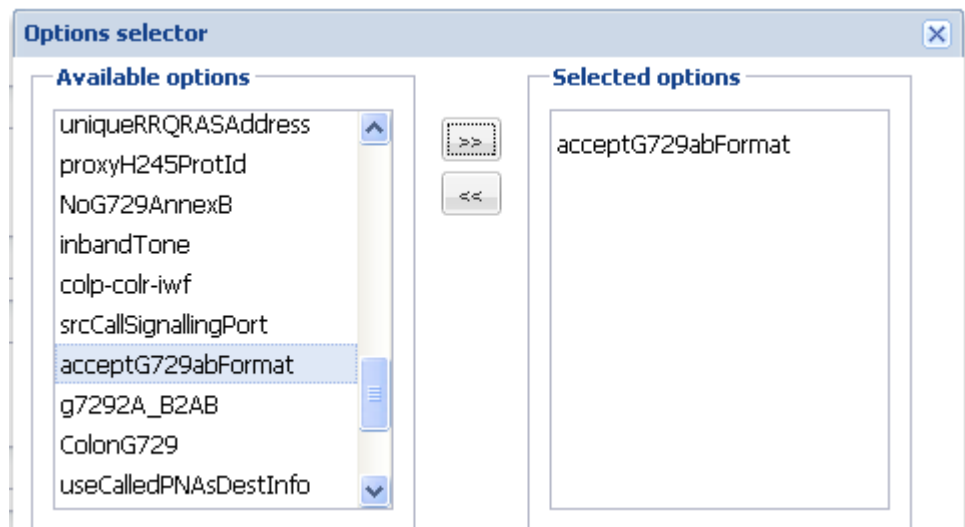
Note: Configuring and applying customized G.729 support will cause the Net-Net SBC to drop in-progress H.323-related calls.

To enable customized G.729 support for IWF calls:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. Scroll to Options and click **Options**.
5. Click **acceptG729abFormat** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
7. Click **OK**.
8. Click **Apply**.

IWF Codec Renegotiation for Audio Sessions

For calls requiring interworking between SIP and H.323, there can be several instances for audio sessions when a mid-call codec change is necessary. These are some examples of when the codec used for voice transportation is necessary:

- Sessions between analog FAX machines that start as regular voice calls but then must use a codec that is fax-signaling tolerant (for example, transparent G.711) when FAX tones are detected; detection takes place after the call has been answered. The case of modem calls is similar.
- An established call is redirected in one carrier's network either to a different enduser or to a media server. In this case, the party to which the call is redirected might not support the codec used in the redirection. If request for a codec change is carried out at the signalling level, the call can proceed with the party to which the call was redirected.
- Endusers might want to change codecs when they suffer low voice quality.

Both SIP and H.323 provide mechanisms for changing codecs during a call: SIP uses the ReINVITE, and H.323 uses the H.245 Request Mode. Using the option called **processRequestModeForIWF** either in an H.323 interface (stack) or an H.323 session agent configuration, you can enable the Net-Net SBC to interwork SIP ReINVITE and H.245 Request Mode requests.

Codec Request Change from the SIP Side

When a SIP party requests a code change, the Net-Net SBC communicates with the H.323 endpoint to renegotiate support for an updated codec. In this renegotiation, the Net-Net SBC presents codec for use ordered according to the SIP side's preference and one is selected. Then the Net-Net SBC handles opening of a new logical channel that uses the updated codec, and closes the old logical channel (that uses the now-outdated codec). On the SIP side, the Net-Net SBC sends a 200 OK with the necessary RTP port and codec information for the new logical channel.

Codec Request Change from the H.323 Side

When the Net-Net SBC receives a codec request change on the H.323 side of an IWF call, it sends a Re-INVITE to the SIP endpoint containing new codec and information. The Net-Net SBC uses IP address and port information it has cached for the H.323 side of the call for the Re-INVITE since H.245 Request Mode requests do not have this data. If the IP address and port combination should subsequently change (in an OLC from the H.323 side), the Net-Net SBC handles additional INTVITEs on the SIP side to support the change.

Exceptional Cases

When the relevant option is enabled, the Net-Net SBC can handle properly the following cases of codec change:

- When the H.323 side rejects the request mode change, the Net-Net SBC response to the SIP side with a 488 Not Acceptable. Session description and state remain unchanged, and the call continues using the original session description.
- When the H.323 side does not respond to the request mode change within the timeout limitation, the Net-Net SBC releases the call on both sides.
- When the SIP side does not respond to the ReINVITE within in the timeout limitation, the Net-Net SBC releases the call on both sides.

- When the intersection of codec is empty, the Net-Net SBC rejects the codec change on the SIP side with a 488 Not Acceptable and on the H.323 side with an H.245 RequestModeReject. Session description and state remain unchanged, and the call continues using the original session description.
- If the Net-Net SBC does not receive any of the LogicalChannel request or acknowledgement messages, the Net-Net SBC releases the call on both sides.

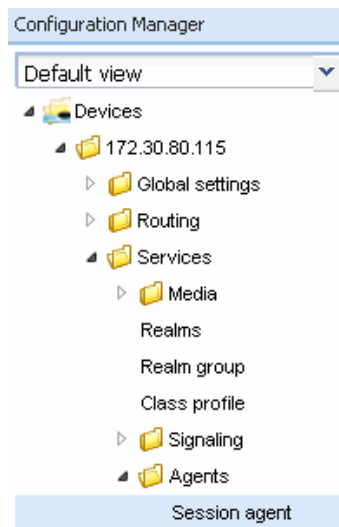
Note that for protocol timeout errors, the preferred behavior is to release the call on both sides. Timeout errors usually indicate network problems, such as an endpoint being unreachable.

Enabling IWF Codec Renegotiation

You can apply the `processRequestModeForIWF=all` to H.323 interfaces (stacks) and to H.323 session agents (sessions agents for which H.323 has been identified in the protocol parameter).

To enable IWF codec renegotiation for an H.323 session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select an H.323 session agent in the table to select it and click **Edit**.
4. Click the **Options** button.
5. Click `processRequestModeForIWF` in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.
7. `processRequestModeForIWF`—Edit the text box to append this option `=all`.

8. Click OK.
9. Click Apply.

SIP Early Media Suppression

This section explains how to configure early media suppression, which lets you determine who can send early media and in what direction. Early media are the RTP/RTCP packets sent from the called party to the caller, or vice versa, before a

session is fully established (before a 200 OK is received). When the Net-Net SBC receives an INVITE message with SDP, it can forward media packets to the calling endpoint as soon as it forwards the INVITE to the next hop. It can also forward media packets received from the calling endpoint to the called endpoint as soon as the Net-Net SBC receives SDP in a SIP response to the INVITE, usually a provisional message. This allows for any early media to be played, such as remote ringback or announcement.

Early media can be unidirectional or bidirectional, and can be generated by the caller, the callee, or both.

With early media suppression, you can block early media until the call is established. You can define which outbound realms or next hop session agents are allowed to send or receive early media. Early media suppression only applies to RTP packets. RTCP packets received by Net-Net SBC are still forwarded to their destination in both directions, unless an endpoint is behind a NAT and the media manager has not been enabled for RTCP forwarding.

Note: To use early media suppression, you cannot configure media release of any kind: same-realm, same-network, or multiple-system media release.

How It Works

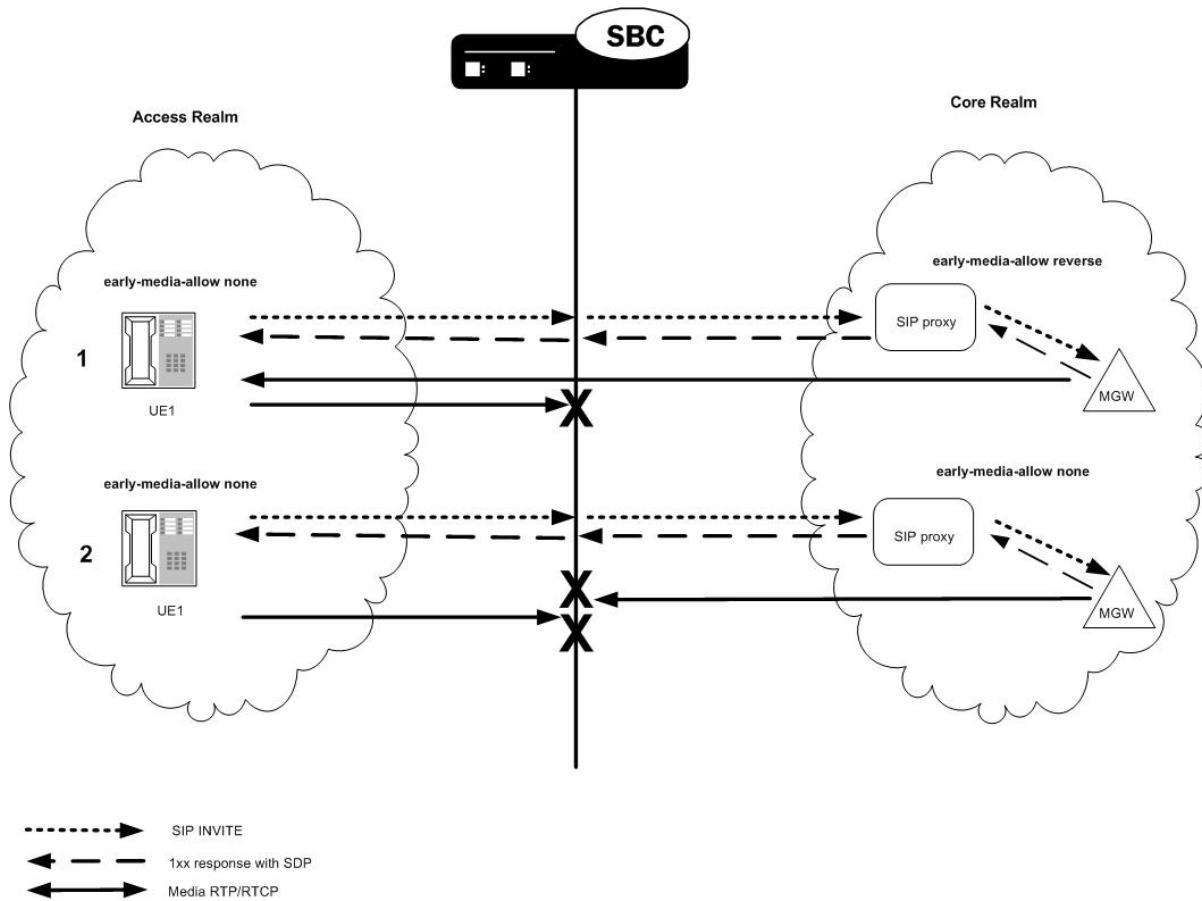
With the SIP-based addressing, early media suppression is based on the outbound SIP interface realms and the value of their early-media-allow parameter. When the Net-Net SBC forwards a SIP INVITE out a SIP interface, the outbound realm is chosen based on the SIP layer information, such as the session agent for the next-hop or the address prefix of the next-hop SIP device. The matching realm's early-media-allow parameter value then applies to either allow all, block all, or block one-way early media until a 200 OK is received. At that point bidirectional media is allowed. The decision is based on SIP-layer addressing of next-hops.

You configure a rule for a realm or a session agent to use early media suppression. An early media suppression rule specifies whether you want to prevent early media in any direction, allow early media going to the calling endpoint in the reverse direction, or allow early media in both directions. The forward direction is when the packets flow from the caller to the called party. The reverse direction is when the packets flow from the called party to the caller.

The early media suppression rule is applied to a session. When the Net-Net SBC initiates a new session, it first checks whether the next hop is a session agent and if so, whether an early media suppression rule has been configured it. If an early media suppression rule is found, the Net-Net SBC enforces it. If the next hop is not a session agent or no early media suppression rule is configured, the Net-Net SBC checks whether an early media suppression rule has been configured for the outbound realm. If it finds one, it enforces it.

Example

The following illustration shows two examples of early media suppression.



1. Caller UE1 makes a call to the PSTN media gateway (MGW). The INVITE traverses from UE1 to the Net-Net SBC through the softswitch to the MGW. The Net-Net SBC allows early media from the core to reach UE1.
2. The PSTN MGW makes a call to UE1. The INVITE traverses to the Net-Net SBC and to UE1. The Net-Net SBC blocks all early media to and from UE1 until a 200 OK is received.

Early Media Suppression Support

The Net-Net SBC supports suppressing early media in the following directions no matter which side makes the SDP offer, until it receives 200 OK for an INVITE:

- Forward direction based on the outbound realm or next-hop session agent
- Forward and reverse directions based on the outbound realm or next-hop session agent.

The Net-Net SBC allows all media when a 200 OK response is received for the INVITE, regardless of whether the 200 OK response contains SDP.

Call Signaling

The Net-Net SBC media manager performs early media suppression according to an early media suppression rule. No change has been made to call signaling. For SIP, the Net-Net SBC still forwards SDP received in an INVITE request or response after performing a NAT to the media connection address. After which, the Net-Net SBC

is ready to receive media packets from the endpoints. If an early media suppression rule has been configured, the Net-Net SBC drops the packets going in the direction being specified by the rule.

For a H.323 to SIP call, early media suppression rule does not change how the Net-Net SBC performs H.225/Q.931 call signaling and starts the H.245 procedure (if required) to establish logical channels for early media on the H.323 leg of the call.

Suppression Duration

When early media suppression is enabled in a session, the block lasts until the session is established. For a SIP to SIP call or an H.323 to SIP call, a session is established when the Net-Net SBC receives a 200 OK response to the INVITE. A 200 OK response to the INVITE terminates early media suppression, even when it does not contain a SDP. (A 200 OK response to a PRACK or an UPDATE request does not terminate early media suppression.) After a session is established, the Net-Net SBC can receive a change in media session (for example, a re-INVITE with a new SDP) without an early media suppression rule blocking the media.

About the Early Media Suppression Rule

An early media suppression rule is configured in the form of a permission. It specifies whether early media is allowed in both directions, the reverse direction only or not at all. Reverse direction media is media sent in the upstream direction towards the calling endpoint.

Session Agent Rule

The next-hop session agent's early media suppression rule is applied regardless of whether the media packet's source or destination address is the same as the session agent's address. For example, if the session's next hop session agent is 10.10.10.5 but the SDP in a 183 response specifies 10.10.10.6 as its connection address.

Rule Resolution

When the call's next hop is a session agent and both the outbound realm of the call and the session agent have an early media suppression rule, the session agent's early media suppression rule takes precedence. If the session agent's early media suppression rule has not been configured, the outbound realm's early media suppression rule is used, if configured.

Selective Early Media Suppression

Normally, the Net-Net SBC performs early media blocking based on destination realm. Calls to such realms are prohibited from sending and receiving RTP until a SIP 200 OK response is received, and you can set the direction of the blocked media.

While decisions to block early media are customarily based on SIP-layer addressing, there are cases when the Net-Net SBC can reject early media based on the SDP address in the SDP answer for a 1XX or 2XX response. By comparing the SDP address with the realm prefix or additional prefix address, it can block early media for matching realms. For these cases, you define global or signaling realms—ones that are not tied to SIP interfaces, but which establish additional address prefixes and rules for blocking early media.

This way, the Net-Net SBC blocks all early media for SIP interface realms, but can accept it for global realms that reference media or PSTN gateways. This configuration allows early media for calls destined for the PSTN, and blocks it for user-to-user and PSTN-to-user calls.

Selective early media suppression addresses the fact that some service providers need to allow early media for certain user-to-user and PSTN-to-user calls to support, for example, custom ringback tones. The enhancements also address the fact that Net-Net SBCs can themselves lose the ability to decide whether or not early media should be blocked when confronted with hairpinned call flows, or with traffic that traverses multiple Net-Net SBCs.

How It Works

To address this need, you can configure realm groups. Realm groups are sets of source and destination realms that allow early media to flow in the direction you configure. For example, you can set up realm groups to allow media from PSTN realms to user realms so that users can listen to PSTN announcements, but prohibit early media from user realms to PSTN realms.

Note that the source and destination realms you add to your lists need to be a global or signaling realm matching the caller's SDP address prefix or a SIP realm.

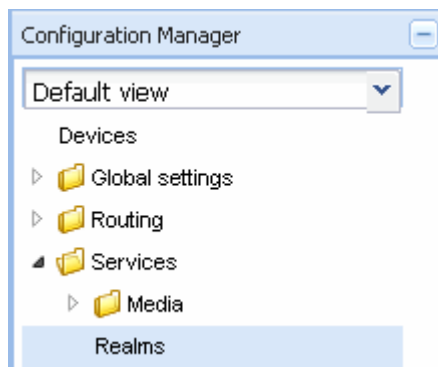
Configuring Early Media Suppression

If the Net-Net SBC is serving as a media bridge, see *Configuring Media Suppression for Media Bridge* for information.

Configuring the Realm

To configure the realm:

1. Expand the Configuration Manager slider -> Services.



2. Click Realms. The Realms table appears in the content area.
3. Select the realm you would like to use and click **Edit**.
4. **Early media allow**—Select the early media suppression rule for the realm from the combination box. The values are:
 - **none**—No early media is allowed in either direction
 - **both**—Early media is allowed in both directions
 - **reverse**—Early media received by Net-Net SBC in the reverse direction is allowed

There is no default value. If you leave this parameter blank, early media is allowed in either direction.

Early media allow

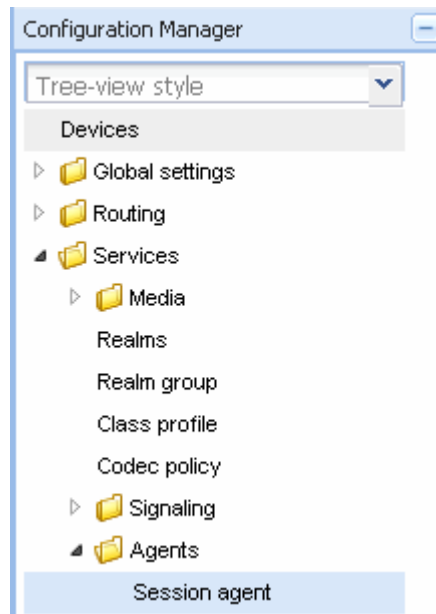
5. Click **Apply**.

Configuring Session Agents

If you do not configure early media suppression for a session agent, the early media suppression for the outbound realm is used, if configured.

To configure session agents:

1. Expand the Configuration Manager slider -> Services -> Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the session agent you would like to use and click **Edit**.
4. **Early media allow**—Select the early media suppression rule for the session agent from the combination box. The values are:
 - **none**—No early media is allowed in either direction
 - **both**—Early media is allowed in both directions
 - **reverse**—Early media received by Net-Net SBC in the reverse direction is allowed

There is no default value. If you leave this parameter blank, early media is allowed in either direction.

Early media allow

5. Click **Apply**.

IWF Codec Renegotiation for Audio Sessions

1976

For calls requiring interworking between SIP and H.323, there can be several instances for audio sessions when a mid-call codec change is necessary. These are some examples of when the codec used for voice transportation is necessary:

- Sessions between analog FAX machines that start as regular voice calls but then must use a codec that is fax-signaling tolerant (for example, transparent G.711) when FAX tones are detected; detection takes place after the call has been answered. The case of modem calls is similar.
- An established call is redirected in one carrier's network either to a different enduser or to a media server. In this case, the party to which the call is redirected might not support the codec used in the redirection. If request for a codec change is carried out at the signalling level, the call can proceed with the party to which the call was redirected.
- Endusers might want to change codecs when they suffer low voice quality.

Both SIP and H.323 provide mechanisms for changing codecs during a call: SIP uses the ReINVITE, and H.323 uses the H.245 Request Mode. Using the option called **processRequestModeForIWF** either in an H.323 interface (stack) or an H.323 session agent configuration, you can enable the Net-Net SBC to interwork SIP ReINVITE and H.245 Request Mode requests.

Codec Request Change from the SIP Side

When a SIP party requests a code change, the Net-Net SBC communicates with the H.323 endpoint to renegotiate support for an updated codec. In this renegotiation, the Net-Net SBC presents codec for use ordered according to the SIP side's preference and one is selected. Then the Net-Net SBC handles opening of a new logical channel that uses the updated codec, and closes the old logical channel (that uses the now-outdated codec). On the SIP side, the Net-Net SBC sends a 200 OK with the necessary RTP port and codec information for the new logical channel.

Codec Request Change from the H.323 Side

When the Net-Net SBC receives a codec request change on the H.323 side of an IWF call, it sends a Re-INVITE to the SIP endpoint containing new codec and information. The Net-Net SBC uses IP address and port information it has cached for the H.323 side of the call for the Re-INVITE since H.245 Request Mode requests do not have this data. If the IP address and port combination should subsequently change (in an OLC from the H.323 side), the Net-Net SBC handles additional INTVITEs on the SIP side to support the change.

Exceptional Cases

When the relevant option is enabled, the Net-Net SBC can handle properly the following cases of codec change:

- When the H.323 side rejects the request mode change, the Net-Net SBC response to the SIP side with a 488 Not Acceptable. Session description and state remain unchanged, and the call continues using the original session description.

- When the H.323 side does not respond to the request mode change within the timeout limitation, the Net-Net SBC releases the call on both sides.
- When the SIP side does not respond to the ReINVITE within in the timeout limitation, the Net-Net SBC releases the call on both sides.
- When the intersection of codec is empty, the Net-Net SBC rejects the codec change on the SIP side with a 488 Not Acceptable and on the H.323 side with an H.245 RequestModeReject. Session description and state remain unchanged, and the call continues using the original session description.
- If the Net-Net SBC does not receive any of the LogicalChannel request or acknowledgement messages, the Net-Net SBC releases the call on both sides.

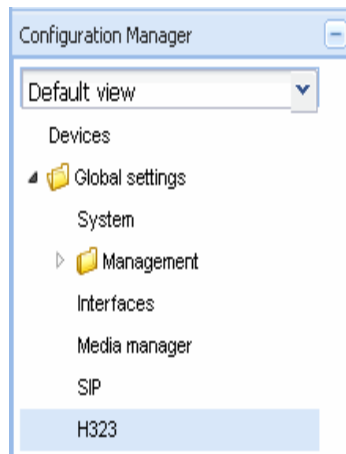
Note that for protocol timeout errors, the preferred behavior is to release the call on both sides. Timeout errors usually indicate network problems, such as an endpoint being unreachable.

Enabling IWF Codec Renegotiation

You can apply the `processRequestModeForIWF=all` to H.323 interfaces (stacks) and to H.323 session agents (sessions agents for which H.323 has been identified in the protocol parameter).

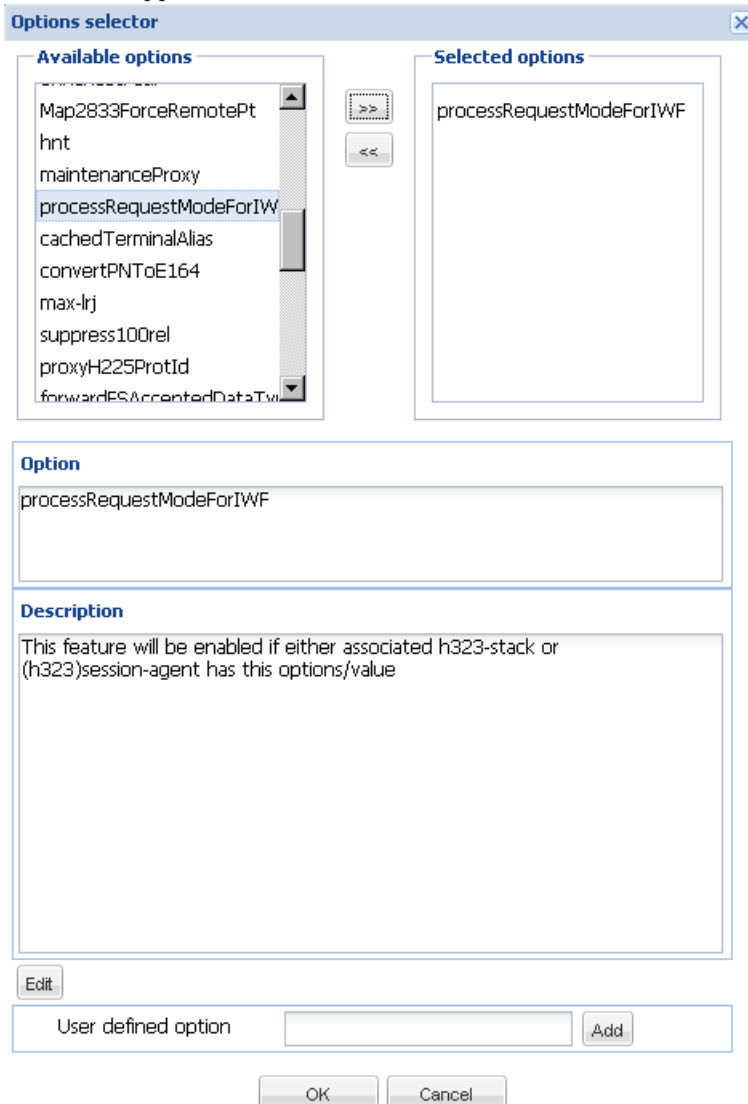
To enable IWF codec renegotiation for an H.323 session agent:

1. Expand the Configuration Manager slider -> Global settings.
2. Click H323.



3. Click the **Options** button. The Options window appears.
4. Click `processRequestModeForIWF` in the Available options list to select it.

5. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



The dialog box is titled "Options selector" and contains two main sections: "Available options" and "Selected options".

Available options: A list box containing the following options: Map2833ForceRemotePt, hnt, maintenanceProxy, processRequestModeForIWF (highlighted), cachedTerminalAlias, convertPNTToE164, max-lrj, suppress100rel, proxyH225ProtId, and forwardESAcceptedDataTvl. Between the two lists are two buttons: ">>" and "<<".

Selected options: A list box containing the option: processRequestModeForIWF.

Option: A text field containing the text: processRequestModeForIWF.

Description: A text area containing the text: This feature will be enabled if either associated h323-stack or (h323)session-agent has this options/value.

Edit: A button located below the description.

User defined option: A text field with the placeholder text "User defined option" and an "Add" button to its right.

OK and Cancel: Two buttons at the bottom of the dialog.

6. **User defined option**—Edit the text box to append this option =all..

User defined option

7. Click Add.
8. Click Apply.

Reliable Response Interworking

There is no configuration required for this feature. It is enabled automatically.

LMSD SIP to IMS SIP Interworking Enhancement

There is no configuration required for this feature. It is enabled automatically.

Introduction

This chapter explains how to configure the Net-Net SBC to support Media Gateway Control Protocol (MGCP/NCS) signaling services.

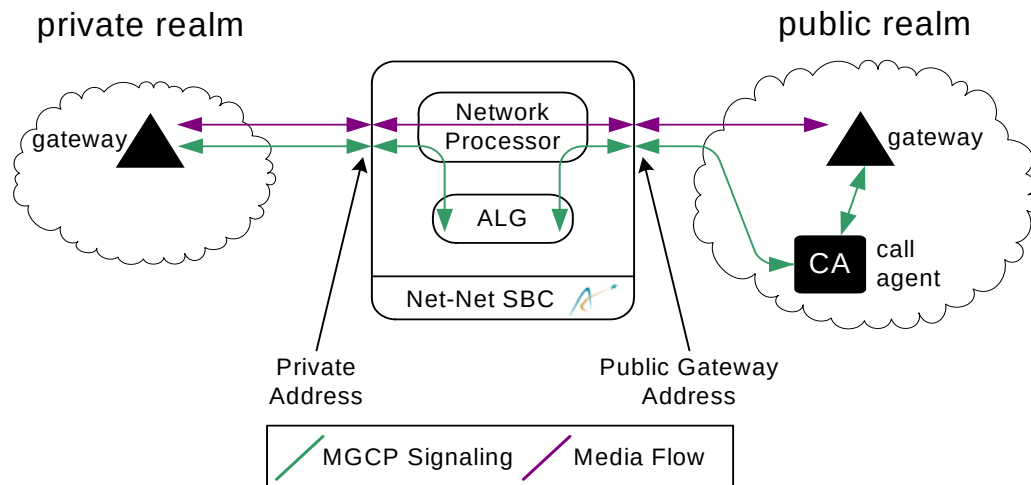
MGCP/NCS Overview

The Net-Net SBC provides MGCP/NCS Application Layer Gateway (ALG) functionality for MGCP/NCS messages between media gateways and media gateway controllers. For the purpose of this document, there are two major types of elements in an MGCP/NCS deployment, endpoints and call controllers. Endpoints encompass Integrated Access Devices (IAD), Multimedia Terminal Adapters (MTA), and Gateways (GW). Call controllers encompass Media Gateway Controllers (MGC), Softswitches, and Call Agents (CA). Throughout this chapter, GW and CA will be used to refer to their respective class of devices. The Net-Net SBC is positioned in the path of packets flowing between media gateways and media gateway controllers to provide a level of packet translation, without directly participating in the media flow.

GWs and CAs are not aware of the role of the Net-Net system; they assume a direct connection to each other. To CAs, the Net-Net system appears to be a GW. To GWs, the Net-Net system appears to be a CA. The Net-Net system, positioned as an intermediary device, provides seamless NAT and packet forwarding functionality.

When MGCP/NCS signaling messages traveling between CAs and GWs pass through the Net-Net system, the Net-Net system identifies the packets as addressed to CAs or GWs. The network processor, in turn, sends these packets to the host processor, which performs the MGCP/NCS NAT and returns the packet to the network processor.

The network processor sends the packet on to its next destination. The following diagram shows an abstract representation of an MGCP/NCS network configuration.



MGCP/NCS and Realms

You need to understand how realms are configured for your MGCP/NCS deployment. A private realm in an MGCP/NCS configuration is the access network. This realm is commonly the public Internet where individual GWs exist. The private realm can also be a VPN. The public realm refers to the service provider's backbone network.

MGCP/NCS configurations must be unique on a per-private realm basis. No two MGCP/NCS configuration elements can have the same private realm. Depending on the design of your network: all of MGCP/NCS configuration elements can share the same public realm; some MGCP/NCS configuration elements can share the same public realm and the remainder can have different public realms; all MGCP/NCS configuration elements can have different public realms. For configurations that include multiple public realms, no two of these public realms can include the same public gateway address and public realm pair.

Note: Public and private realms must be configured first in the realm configuration element, as explained in the Realm Configuration section of this guide.

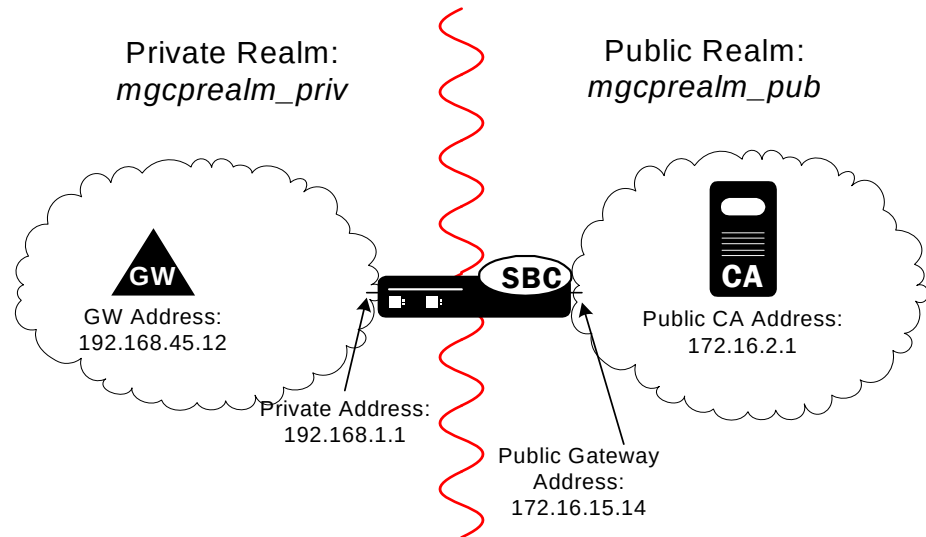
MGCP/NCS NAT Traversal

Net-Net SBCs work seamlessly when GWs in the private realm are located behind NATs. The MGCP/NCS configuration provides a mechanism for NAT traversal. When the Net-Net system recognizes that a layer 3 IPv4 source address does not match the same information provided in the layer 5 MGCP signaling message, NAT traversal is enabled. The Net-Net SBC sends packets through a NAT at a given interval to keep a pinhole open and prevent MGCP/NCS connections from prematurely closing.

The Net-Net SBC recognizes when an endpoint is behind a NAT because of the discrepancy between the host portion of the endpoint ID and the layer 3 source address. This recognition enables the mechanisms for maintaining NAT traversal. You do not have to explicitly enable NAT traversal unless your deployment uses FQDNs in the endpoint ID when behind a NAT. In that case, NAT traversal is not automatically enabled; you must explicitly enable it in this situation.

MGCP/NCS Network Topology

The following network topology map shows three network elements: GW, Net-Net SBC, and CA. There are two logical realms: `mgcprealm_priv` and `mgcprealm_pub`. The GW exists in `mgcprealm_priv` and the CA exists in `mgcprealm_pub`. The GW and CA each have one assigned IPv4 address. The Net-Net SBC has two IPv4 addresses, one on each network interface existing in each of the two realms.



MGCP/NCS Configuration Overview

To create a basic MGCP/NCS configuration, populate the MGCP configuration element as follows.

1. Set the private realm information. This includes the private realm ID, and the IPv4 address and port number of the network interface on which the private realm exists. The respective parameters for these fields are **Private Realm**, **Private Address**, and **Private port**. The private address and port pair compose a virtual CA on this Net-Net SBC from the GWs' points of view.
2. Set the public CA information. This includes the IPv4 address or hostname and port number of the call agent located in the public realm. The respective parameters for these fields are **Public Call Agent Host**, **Public Call Agent Port**, **Public Call Agent Address**.
3. Set the public gateway information. This includes the public realm ID, and the IPv4 address (or hostname) and port number of the network interface on which the public realm exists. In addition, a second public gateway port number can be identified. The public realm is where the CAs and DNS servers exist. The public address and port pair compose a virtual GW on this Net-Net SBC from the CA's point of view. The respective parameters for these fields are **Public Realm**, **Public Gateway Host**, **Public Gateway Address**, and **Public Gateway Port**. A **Second Public Gateway Port** can also be configured.
4. Set the NAT traversal configuration. MGCP/NCS NAT must be explicitly enabled when using FQDNs for endpoint IDs. The parameters for this field are **NAT Traversal** and **Audit Interval**. These configuration elements only refer to enabling NAT traversal for GWs that exist in the private realm.
5. Set the ALG port. This field defines the port associated with this MGCP Configuration element. Each MGCP element must have a unique ALG port so that the ALG process can distinguish which MGCP element (stack) is communicating with the network processor. The parameter defining this field is **ALG Port**.

Before You Configure

In order for the Net-Net SBC to pass media and control traffic for MGCP/NCS from the public realm to the private realm and vice versa, you must set the following elements for a baseline configuration:

- physical interfaces
- network interfaces
- media manager
- steering pools
- realm configurations

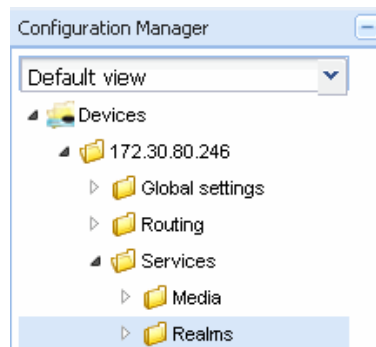
Configuring MGCP/NCS Signaling Services

You can create and configure private and public MGCP realms.

Creating a Private MGCP Realm

To create a private MGCP realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click **Add**. The Add Realm window appears.
4. **Name**—Enter a name for the private MGCP realm.
5. **Sub port**—Select the VLAN identifier from the combination box for the network interface to which you are assigning this realm.
6. **Interface name**—Select a physical interface from the combination box.

 A screenshot of the 'Add Realms' dialog box. It contains three main input fields:

- *Name**: A text box containing 'mgcprealm-priv' with a '(Unique required)' label to its right.
- Network interface**: A section header.
- *Sub port**: A dropdown menu showing '0' with a '(Default:0)' label to its right.
- *Interface name**: A dropdown menu showing 's0p0'.

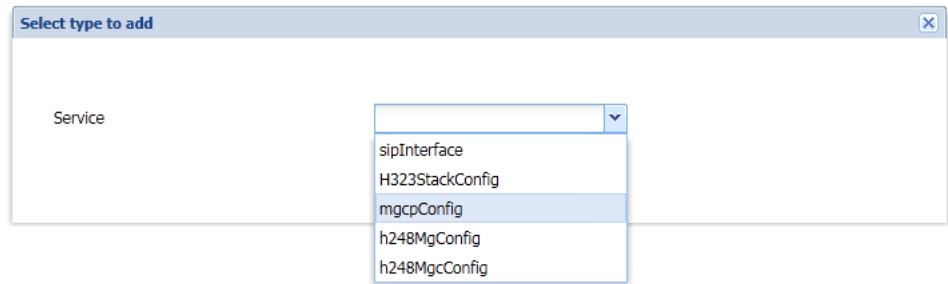
 At the bottom of the dialog are 'Apply' and 'Cancel' buttons.

7. Click **Apply** to continue.

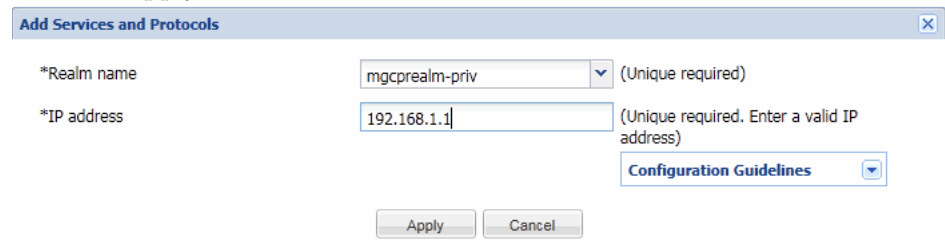
Configuring MGCP ER Service

8. Select the MGCP realm you want to configure in the table and click **Edit**.
9. Select the Services and Protocols tab.

10. Click **Add**. The Add Realm service window appears.
11. **Service**—Select **MGCP-ER** from the combination box.



12. Click **Apply**.
13. Select the mgcpConfig from the Services table and click **Edit**.
14. **Realm**—Select the MGCP realm from the combination box.
15. **IP address**—Enter the IPv4 address of the network interface inside of the private realm.
16. Click **Apply** to continue.



The newly created MGCP service appears in the Services table.

17. Click the MGCP ER service in the table to select it and click **Edit**.
18. **IP port**—Enter the port number of the private realm's network interface.
19. **NAT traversal**—Select enabled from the combination box to enable NAT traversal.
20. **Audit endpoint frequency (sec)**—Enter the interval at which AUEP messages are sent to gateways behind a NAT.

IP port

NAT traversal

Audit endpoint frequency (sec)

21. Click the Backbone realm tab at the top of the content area.
22. **Name**—Enter the public GW host name or the Public GW IP Address. in the MGCP signaling interface section.
23. **IP address/Bit mask**—Enter the public GW IP address bitmask portion if you entered the public GW as an IP address.
24. **IP port-transmit**—Enter the public GW port.

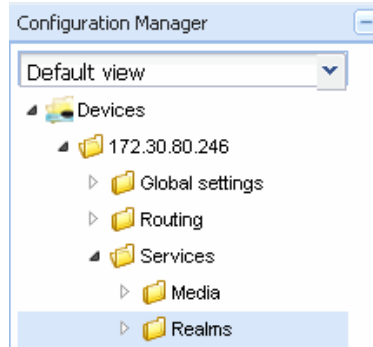
25. **IP port-receive**—Enter the second public GW port if applicable.

Name	mgcp1
IP address/Bit mask	172.162.2.1
IP port-transmit	2427
IP port-receive	0

26. Click **Apply**.

Creating a Public MGCP Realm

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click **Add**. The Add Realm window appears.
4. **Name**—Enter a name for the public MGCP realm.
5. **Sub port**—Select the VLAN identifier from the combination box for the network interface to which you are assigning this realm.
6. **Interface name**—Select a physical interface from the combination box.

 A screenshot of the 'Add Realms' dialog box. It has three input fields:

- *Name: 'mgcprealm-pub' (with '(Unique required)' text to the right)
- *Sub port: '0' (with '(Default:0)' text to the right)
- *Interface name: 's0p0'

 At the bottom are 'Apply' and 'Cancel' buttons.

7. Click **Apply** to continue.

DNS Authentication

This section explains how to configure DNS authentication.

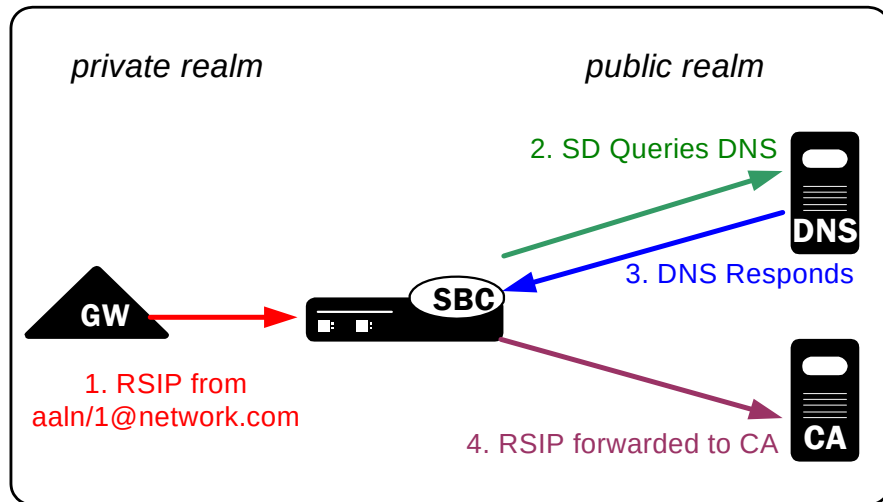
Some MGCP/NCS deployments require DNS authentication of endpoints for security purposes. The Net-Net SBC can perform DNS authentication against endpoint IPv4 addresses. This authentication checks an endpoint's domain name (layer 5) against the endpoint's IPv4 address received in a layer 3 message. If the two identifiers match, an original RSIP is forwarded to the call agent and the phone call can begin.

Note: The endpoint equipment and the DNS server must be compatible with the DNS authentication process.

In the diagram below, the following four steps take place:

1. The endpoint registers with an RSIP to the Net-Net SBC.
2. The Net-Net SBC queries a DNS server to check if the layer 5 endpoint name is the same as the layer 3 IPv4 address of the registering GW.

3. The DNS server responds to the Net-Net SBC indicating that the IPv4 address and domain name match.
4. The Net-Net SBC forwards the RSIP from the GW to the CA.



If the domain name and IPv4 address do not match, the Net-Net SBC issues an authentication failed 400 message back to the endpoint and the RSIP is not forwarded to the call agent.

DNS Authentication Configuration Prerequisites

You need to configure the following elements for DNS authentication to work:

- Translation rules—unique sets of rules applied to incoming and outgoing calls. Translations are used to add, delete, or change character strings within an address.
- Session Translation—defines how translation rules are applied to both incoming and outgoing calls. Multiple translation rules can be referenced and applied using the Session Translation element, which groups rules together and allows them to be referenced by one name.

The Net-Net SBC applies the translation rules established in this field in the order in which they are entered. To enable DNS authentication without using any session translations, set up an empty session translation. (See the Number Translation section to learn how to configure a translation rules and session translation elements.)

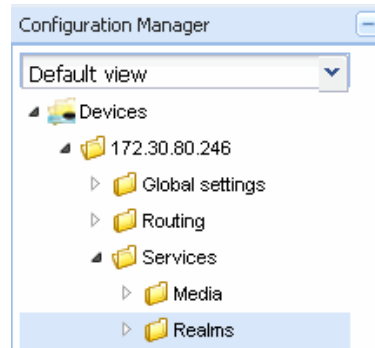
- DNS servers' IPv4 addresses must also be configured. These parameters are found in the network interface elements. You can configure as many as three DNS servers per network interface. Instructions on how to add DNS servers is located in the Network Interface section of this guide.

After the translation rules and sessions translation prerequisites have been set, set the parameters that enable DNS authentication for your MGCP/NCS configuration.

Configuring DNS Authentication

To configure DNS authentication:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Scroll to the MGCP endpoint profile section.
7. **DNS translation**—Select a translation profile from the combination box.
8. **DNS authentication**—Select enabled from the combination box.

NAT mode	LineUnit
Divisor for computing endpoint name	256
Unit prefix for creating endpoint name	
DNS translation	
DNS authentication	enabled

9. Click **Apply**.

Endpoint Profile

An MGCP/NCS deployment can take advantage of various ways of representing endpoints to the call agent. Selecting the proper endpoint representation mode can solve the need for unique endpoint representation when endpoints are located behind NATs. Specialized endpoint representation can also be used for formatting endpoints for different call agent systems.

In a common VoIP scenario, multiple GWs with unique private IPv4 addresses exist behind a NAT. The identifiers of two devices might be aaln/1@192.168.200.11 and aaln/1@192.168.200.12 respectively. After packets from these two devices leave the NAT and enter public address space, bound for a Net-Net SBC, they both have the same endpoint ID. Thus, there is no way to distinguish traffic sent from or destined to each device. Traffic from both phones could look like aaln/1@63.168.127.12 if

63.168.127.12 was the public IPv4 address of the NAT. To distinguish traffic, the Net-Net SBC can create unique endpoint names.

Creating unique endpoint names involves reformatting the endpoint name of a GW. A unique identifier is calculated for each endpoint and then inserted into the full endpoint name that remains intact after traversing a NAT. The unique identifier is calculated by using the mode and divisor fields, which are compliant with MGCP/NCS specifications.

Endpoint Number Computation

The Net-Net SBC creates an endpoint identifier to insert into the layer-5 name of an endpoint when an appropriate mode option is chosen. Endpoint identifiers are computed differently if a GW is behind a NAT or not. In the following examples, the divisor is 65536.

Endpoint Not Behind a NAT

The computation of an endpoint identifier for an IPv4 address not behind a NAT is as follows:

1. Multiply the decimal equivalent of the first octet of the IPv4 address by 256^3 , see column A in the following example.
2. Multiply the decimal equivalent of the second octet of the IPv4 address by 256^2 , see column B in the following example.
3. Multiply the decimal equivalent of the third octet of the IPv4 address by 256^1 , see column C in the following example.
4. Multiply the decimal equivalent of the fourth octet of the IPv4 address by 256^0 , see column D in the following example.
5. Add the numbers (A through D) computed in steps 1-4.
6. Calculate the modulus on the number computed in step 5 by the number chosen for the divisor (see next section) parameter. The result is the unique endpoint identifier.

GW not behind a NAT:

IAD IP Address:	192	168	45	12
	x	x	x	x
	256^3	256^2	256^1	256^0
	A	B	C	D

$(A+B+C+D) \% \text{divisor} = \text{endpoint identifier}$

$3221225472 + 11010048 + 11520 + 12 = 3232247052$

$3232247052 \% 65536 = 11532$

Endpoint behind a NAT

The computation of an endpoint identifier for an IPv4 address behind a NAT is as follows:

1. Multiply the decimal equivalent of the third octet of the public IPv4 address by 256^3 , see column A in the following example.
2. Multiply the decimal equivalent of the fourth octet of the public IPv4 address by 256^2 , see column B in the following example.

3. Multiply the decimal equivalent of the third octet of the IPv4 address behind the NAT by 256^1 , see column C in the following example.
4. Multiply the decimal equivalent of the fourth octet of the IPv4 address behind the NAT by 256^0 , see column D in the following example.
5. Add the numbers (A through D) computed in steps 1-4.
6. Calculate the modulus on the number computed in step 5 above by the number chosen for the divisor (see next section) parameter. The result is the unique endpoint identifier.

GW behind a NAT

IAD IP Address:	192	168	45	12
			$\times$ 256^1	$\times$ 256^0
			C	D
Gateway IP Address:	10	11	12	13
			$\times$ 256^3	$\times$ 256^2
			A	B

$(A+B+C+D) \% \text{divisor} = \text{endpoint identifier}$

$201326592 + 851968 + 11520 + 12 = 202190092$

$202190092 \% 65536 = 11532$

Valid Divisors

Valid divisors are: 256^1 , 256^2 , 256^3 , 256^4 .

Endpoint Translation Mode

There are seven modes used to translate MGCP/NCS endpoint names. These modes specify the order and format of layer-5 endpoint names. Given an endpoint name entering a Net-Net SBC from the private MGCP realm, translation modes describe the format of the endpoint name as it exits the Net-Net SBC into the public realm.

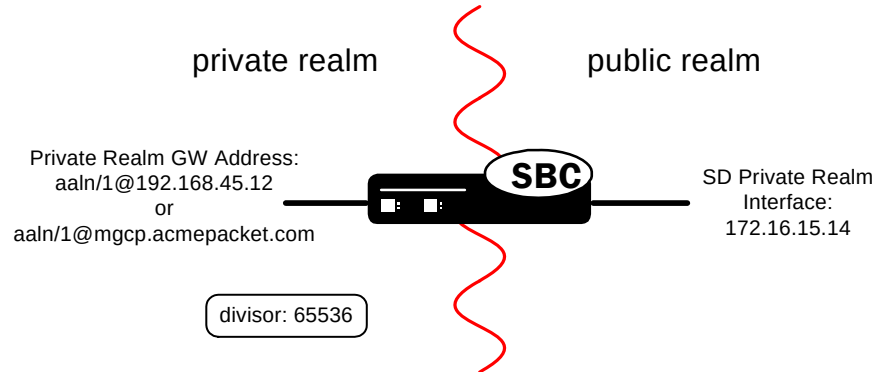
For the following examples, the pre-Net-Net SBC endpoint name is described as: `<endpoint>/<unit>@gateway`, where gateway is either an FQDN or IPv4 address.

Mode	Description
None	The endpoint name will not be translated.
OnlyHost	Equivalent to using the None mode.

Mode	Description
Host	Inserts a term before the unit number in the endpoint name. If the gateway is an FQDN, the left-most part of the FQDN, after the left-most dot, is used as the unit term. If the gateway is an IPv4 address, the decimal equivalent of the IPv4 address is used as the unit term. (The decimal equivalent of an IPv4 address equals the A+B+C+D equation in the previous section). Example: <endpoint>/<unit-term><unit>@post.SBC.IP
LinePrefix	Used primarily in conjunction with a divisor field. This mode inserts the endpoint identifier immediately before the unit number. If the pre-Net-Net SBC endpoint is an FQDN, the most recently received Layer3 IPv4 address is used to compute the endpoint identifier. If the divisor field is left blank, the endpoint identifier will be the decimal equivalent of the IPv4 address. Example: <endpoint>/<endpoint-identifier><unit>@post.SBC.IP
LineUnit	Used primarily in conjunction with a divisor field. This mode inserts the endpoint identifier immediately before the unit number. A slash is inserted between the endpoint identifier and the unit number. If the pre-Net-Net SBC endpoint is an FQDN, the most recently received Layer3 IPv4 address will be used to compute the endpoint identifier. If the divisor field is left blank, the endpoint identifier is the decimal equivalent of the IPv4 address. Example: <endpoint>/<endpoint-identifier>/<unit>@post.SBC.IP
FQDN	The FQDN is surrounded by slashes and inserted between the endpoint and unit number. In addition, the dots are removed from the FQDN. If the endpoint is identified by IPv4 address, its decimal equivalent is inserted in the FQDN position. Example: <endpoint>/<FQDNwithoutdots>/<unit>@post.SBC.IP
FQDN2	The FQDN is inserted between the endpoint and unit number with slashes on either side of it. Note that the difference between FQDN2 and FQDN modes is that the dots are NOT removed from the FQDN in FQDN2 mode. If the endpoint is identified by IPv4 address, its decimal equivalent is inserted in the FQDN position. Example: <endpoint>/<FQDN>/<unit>@post.SBC.IP

Endpoint Translation Examples

In the following figure, all modes and address representations are described according to the mode chosen.



Endpoint Translation

In the following examples, the divisor is 65536. Therefore, the endpoint identifier is 11532.

Mode	FQDN Addressing	IP Addressing	
None	aaln/1@mgcp.acmepacket.com	aaln/1@192.168.45.12	Pre-SD
	aaln/1@mgcp.acmepacket.com	aaln/1@192.168.45.12	Post-SD
Host	aaln/1@mgcp.acmepacket.com	aaln/1@192.168.45.12	Pre-SD
	aaln/mgcp1@172.16.15.14	aaln/115321@192.168.45.12 aaln/7557726841@192.168.45.12 *	Post-SD
LinePrefix	aaln/1@mgcp.acmepacket.com	aaln/1@192.168.45.12	Pre-SD
	aaln/115321@172.16.15.14	aaln/115321@172.16.15.14	Post-SD
	aaln/7557726841@192.168.45.12 *	aaln/7557726841@192.168.45.12 *	
LineUnit	aaln/1@mgcp.acmepacket.com	aaln/1@192.168.45.12	Pre-SD
	aaln/11532/1@172.16.15.14	aaln/11532/1@172.16.15.14	Post-SD
	aaln/7557726842/1@172.16.15.14 *	aaln/7557726842/1@172.16.15.14 *	
FQDN	aaln/1@mgcp.acmepacket.com	aaln/1@192.168.45.12	Pre-SD
	aaln/mgcpacmepacketcom/1@172.16.15.14	aaln/755772684/1@192.168.45.12	Post-SD
FQDN2	aaln/1@mgcp.acmepacket.com	aaln/1@192.168.45.12	Pre-SD
	aaln/mgcp.acmepacket.com/1@172.16.15.14	aaln/755772684/1@192.168.45.12	Post-SD

* denotes that the divisor parameter was not configured.

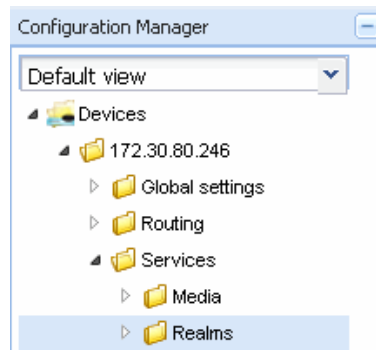
Unit Prefix

You can configure a prefix string on the unit term in an MGCP address. This prefix is configured with the unit-prefix parameter in the MGCP configuration element. A unit-prefix can be used as a tag to mark all calls originating in a specific realm. Unit prefixes are used primarily for routing purposes.

Configuring MGCP Endpoint Profile

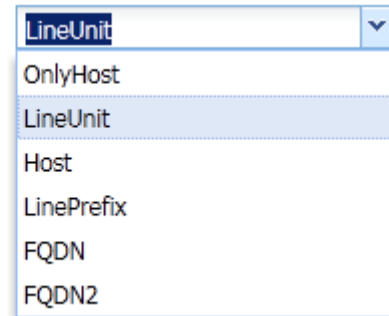
This section describes how to configure endpoint translation settings on your Net-Net SBC from Net-Net Central.

1. Expand the Configuration Manager slider > Services.



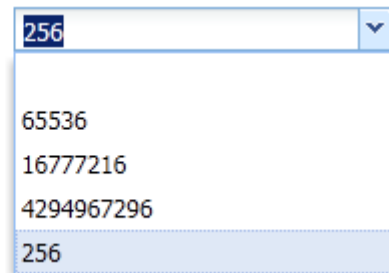
2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Scroll to the MGCP endpoint profile section.
7. **NAT mode**—Choose the mode from the combination box.

NAT mode



8. **Divisor for computing endpoint name**—Choose a divisor from the combination box. Divisors are given as classful addresses.

Divisor for computing endpoint name



9. **Unit prefix for creating endpoint name**—Enter the string to prepend to the unit term in endpoint names.

Unit prefix for creating endpoint name

10. **DNS translation**—Leave this combination box blank.

11. **DNS authentication**—Leave this parameter disabled.

NAT mode	LineUnit	▼
Divisor for computing endpoint name	256	▼
Unit prefix for creating endpoint name	untiprefix	
DNS translation		▼
DNS authentication	disabled	▼

12. Click **Apply**.

MGCP/NCS Options

This section describes the MGCP/NCS features. The MGCP/NCS features are:

- Send Media Only
- X-Via Header Configuration

You configure these features using the MGCP/NCS options parameter.

MGCP Options

The following table lists the MGCP options supported by the Net-Net SBC.

Option	Description
alg-aging-max-loop	Maximum number of endpoints examined for every controlled timer tick (two seconds) to determine aging
alg-again-max-exp	Maximum number of endpoints the Net-Net SBC will age per loop, limits a flood of endpoints from being aged at the same time
alg-aging-cpu-limit	Threshold for percentage of CPU; if CPU utilization exceeds this threshold, alg-aging-load-lax-loop and alg-aging-load-max-exp are used
alg-aging-load-max-loop	When CPU threshold has been exceeded, maximum number of endpoints examined for every controlled timer tick (two seconds) to determine aging
alg-aging-load-max-exp	When CPU threshold has been exceeded, acceptable maximum number of endpoints the Net-Net SBC will age per loop, limits a flood of endpoints from being aged at the same time
algd-load-average-number-periods	The Net-Net SBC's MGCP task incorporates the average CPU utilization into its decision to enter congestion state (and start sending 403 messages to new NTFY off-hook commands). The Net-Net SBC computes average CPU utilization over the period of time equal to algd-load-measurement-interval multiplied by the algd-load-average-number-periods value. The MGCP task enters congestion state when both the current CPU utilization and the average CPU utilization exceed the algd-load-limit value.
algd-load-measurement-interval	Controls how often the Net-Net SBC's MGCP task checks the CPU utilization of application tasks. A small interval makes the task respond more quickly to changes in CPU utilization, but adds overhead to the CPU utilization of the task individually and to the overall CPU utilization of application tasks.
algd-load-minimum-rejection-rate	Specifies the minimum rejection rate when computed using the algd-load-reduction-pct value (when you define the algd-load-reduction-pct). If a new rejection rate is less than the value set using this option, then the Net-Net SBC sets the new rejection rate to 0. The default is 5.

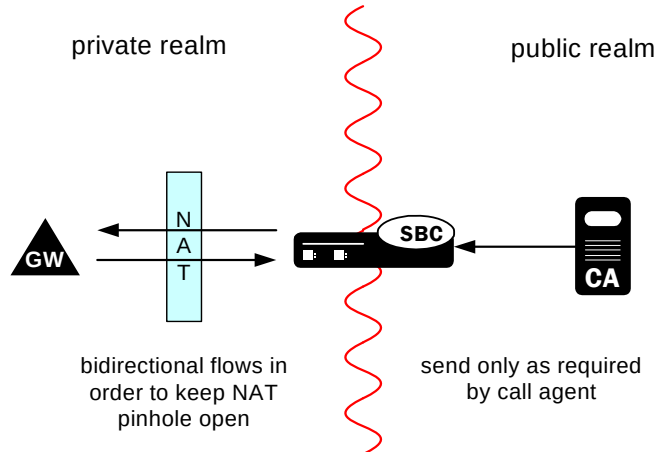
Option	Description
algd-load-reduction-pct	Computes a new rejection rate if there is a decrease in CPU utilization during congestion state and standby state. The new rejection rate equals 100 minus the algd-load-reduction-pct value, as a percentage of the previous rejection rate. For example, if the previous rejection rate is 10% and the algd-load-reduction-pct value is 10%, then the new rejection rate is 90% of 10%, which is 9%. The valid range is 1 to 100, with a default of 0.
algd-load-standby-period	Specifies the time in seconds that the Net-Net SBC stays in a standby state after CPU utilization has receded below the load limit and before it goes back to the normal state. If overload conditions return, the Net-Net SBC returns to the overload state. The valid range is 1 to 60 seconds, with a default of 60.
drain-sendonly	Ensures that send-only devices never receive media traffic. When a gateway tries to contact a send-only MGCP/NCS element through a Net-Net SBC, the Net-Net SBC stops the gateway's traffic from reaching the send-only device. This commonly happens in a hosted NAT traversal situation. At the same time, a gateway located behind a NAT is not treated as a send-only device.
ep-inactivity-timer	Sets the amount of time in seconds before an MGCP endpoint expires. To stagger deletions, the ep-inactivity-timer-rate value can be added to the inactivity timer, if multiple endpoints send messages within the inactivity timer rate window.
ep-inactivity-timer-rate	Defines a minimum gap in milliseconds between timers so that multiple expiring timers do not cause a CPU utilization spike. If you do not set a time, the Net-Net SBC uses a 100 millisecond default.
super-nat	Enables a Net-Net SBC configured to use the masquerading feature (MGCP/NCS 1:1 IPv4 Address Mapping for Gateways and Endpoints) to pass any message coming from the call agent to the gateway, even when the Net-Net SBC does not have an entry for that gateway.
x-keep-alive	The Net-Net SBC's default method of performing Hosted NAT Traversal (HNT) in MGCP/NCS environments is to send periodic AUEP messages to each endpoint. If it is sent prior to the NAT binding time-to-live (TTL) expiry time, this traffic refreshes the binding on the intermediary NAT device. When the Net-Net SBC receives an RSIP with an X-Keepalive header present, it checks to see if that endpoint is known. • If it is known, the Net-Net SBC responds with a 200 OK directly and suppresses the AUEP keepalives that are generated. • If it is not known, the Net-Net SBC replaces the X-Keepalive header with Restart and forwards it to the call agent.

Option	Description
x-useraddress	Enables IP address carrying, which allows multiple active registrations for the same user at different locations.
x-via=<endpoint>, <both>	To configure MGCP/NCS X-Via header for 911 services, set the x-via options parameter as follows: - options x-via=endpoint—This option signifies that the endpoint is either a router or a phone. The X-via header uses a GW's IPv4 address when GWs have public IPv4 addresses. The X-via header uses the NAT's public address when GWs use private addressing from behind a NAT. - options x-via=both—This option signifies that there are two addresses inserted into the X-via header, the private IPv4 address of the GW and the public IPv4 address of the Net-Net SBC.

Send Media Only

In an MGCP/NCS deployment, network elements like announcement servers and media servers may be send-only devices in that they can never receive network traffic. If these send-only devices receive traffic, they might behave improperly causing unstable network conditions.

The MGCP/NCS send media only feature ensures that send-only devices never receive media traffic. When a gateway tries to contact a send-only MGCP/NCS element through a Net-Net SBC, the Net-Net SBC stops the gateway's traffic from reaching the send-only device. This commonly happens in a hosted NAT traversal situation. The Net-Net SBC needs the gateway to send data through the NAT in order to keep the pinhole open. However, if the data coming from the gateway is passed through the Net-Net SBC toward the send only device, undesirable consequences may result.



Enabling the send media only feature on the Net-Net SBC respects the send-only device's need to never receive traffic. At the same time, a gateway located behind a NAT is not treated as a send-only device. The gateway and Net-Net SBC communicate in a way that keeps the NAT pinhole open. All traffic received from the gateway is dropped at the Net-Net SBC and not forwarded toward the real Call Agent.

Signaling the Source IPv4 Address of Endpoints for 911 Services

The MGCP/NCS X-Via header enables the Net-Net SBC to handle 911 class calls. The requisite information is inserted in the MGCP/NCS message in the X-Via header in order that the CA can accept and act on 911 calls.

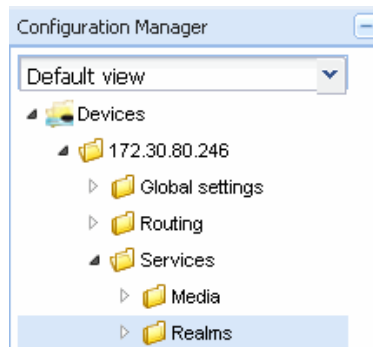
Loose Authentication

When DNS authentication is enabled, you can force the Net-Net SBC to immediately start a call, before the caller is successfully authenticated. This mode works by assuming a successful authentication will be made and immediately sending a NTFY to a CA.

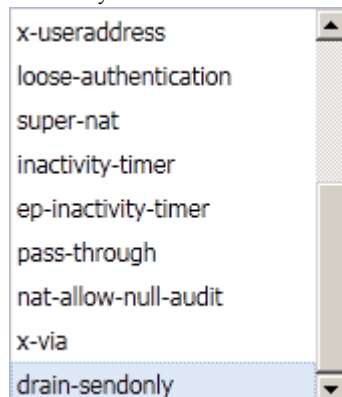
If the DNS query fails completely, for example the query can not reach the DNS server, then the call will still connect. If the DNS query returns an authentication failure, the AUEP (started above) to the endpoint will be stopped and all future NTFY message are not sent to the endpoint.

Configuring MGCP Options

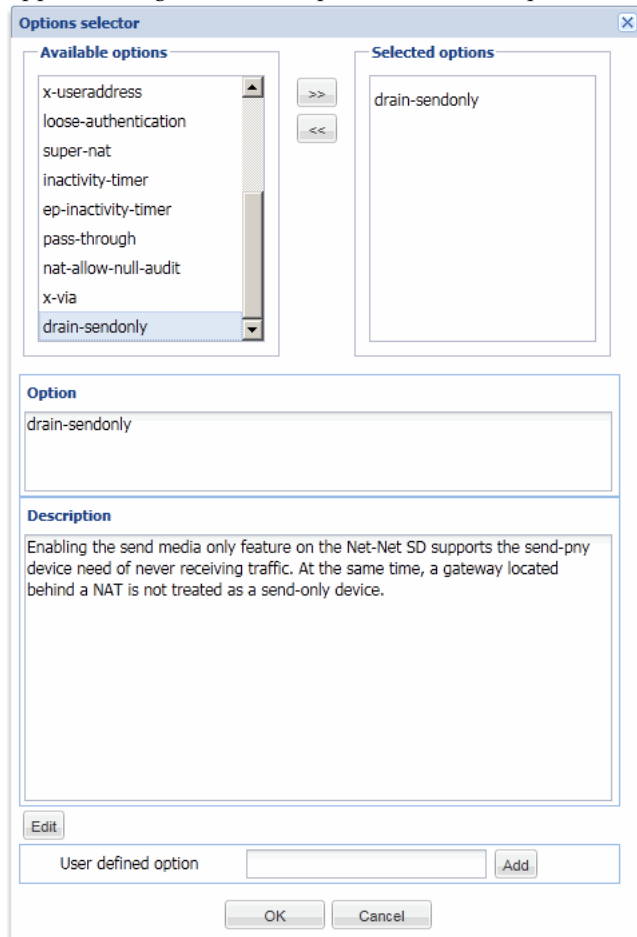
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click **Options** under Options in MGCP. The Options dialog box appears.
7. Click the option name in the Available options list. For example, click drain-sendonly.



8. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.



9. Click **OK**.
10. Click **Apply**.

Configuring the X-Via Header Option

To configure MGCP/NCS X-Via header for 911 services, set the options parameter as follows.

- The X-via header is configured in the options parameter. There are two ways to configure this.
 - options x-via=endpoint**—This option signifies that the endpoint is either a router or a phone. The X-via header uses a GW's IPv4 address when GWs have public IPv4 addresses. The X-via header uses the NAT's public address when GWs use private addressing from behind a NAT.
 - options x-via=both**—This option signifies that there are two addresses inserted into the X-via header, the private IPv4 address of the GW and the public IPv4 address of the Net-Net SBC.

2. Type the option's value in its entry box. For example.

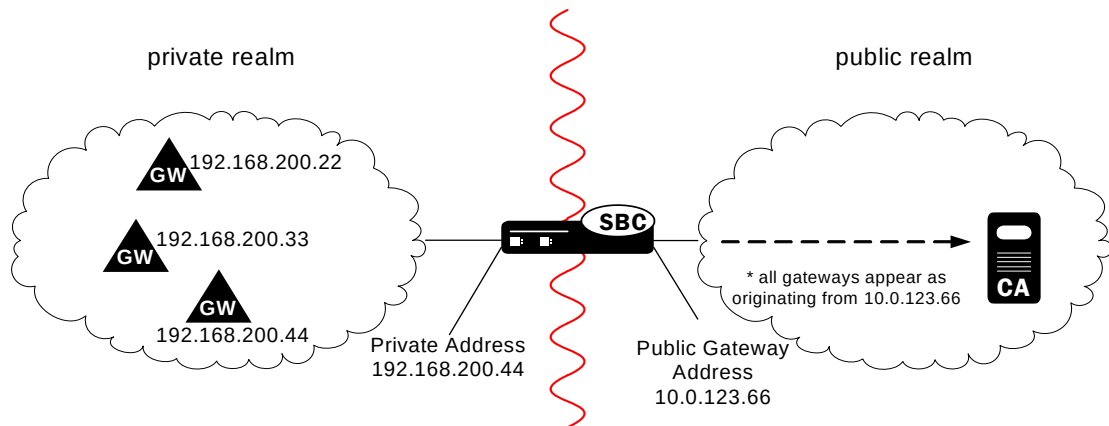
Option
x-via=endpoint

3. Click OK.
4. Click Apply.

MGCP/NCS 1:1 IPv4 Address Mapping

In the typical MGCP/NCS deployment, several GWs that send their traffic through the Net-Net SBC exist in the private realm. These GWs are presented to the CA in the public realm as originating from the public gateway address on the Net-Net SBC. The public gateway address exists on the public-realm-facing network interface as a single IPv4 address and port pair.

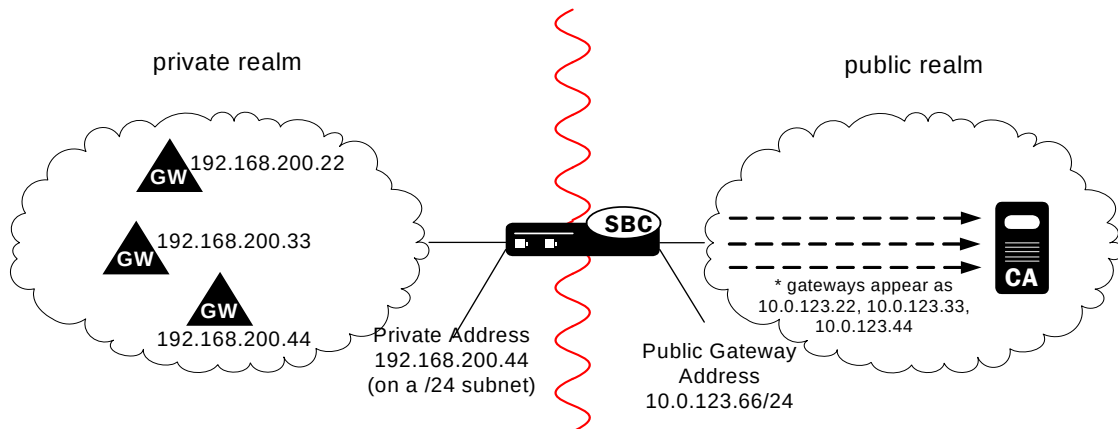
In such a configuration, the GWs believe that the private realm interface on the Net-Net SBC is the CA. Likewise, the CA believes that public realm interface on the Net-Net SBC is one or many GWs. The CA would see all traffic coming from the gateways as sourced from public gateway address. In the following diagram, the CA sees the traffic coming from the GWs as sourced from the public gateway address 10.0.123.66.



From the perspective of the CA, there is only one GW with which it communicates, even though there can be multiple GWs behind the Net-Net SBC.

In some MGCP/NCS deployments, the CA needs to interpret each GW as originating from a distinct IPv4 address. The Net-Net SBC can make each GW appear distinct by assigning it its own unique IPv4 address in the public realm.

The following diagram illustrates that when 1:1 IPv4 address mapping is enabled, all traffic originated in the private realm from the GWs appears with unique IPv4 addresses per gateway in the public realm.

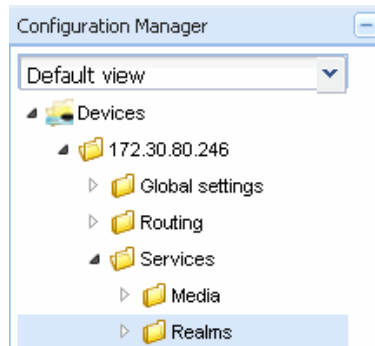


Configuring the netmask on the Public Gateway Address parameter tells the Net-Net SBC how much of the GW's address to copy onto the public realm traffic. In the example, the final 8 bits of each GW is appended to the public gateway address's first 24 bits. This combination is used to present all GWs with unique IPv4 addresses to the CA.

Configuring MGCP/NCS 1:1 IPv4 Address Mapping

When the public gateway address is set to an IPv4 address and netmask combination, all of the host bits (32 minus the netmask) are copied from the incoming source address in the packet to the outgoing source address in the packet. However, this only works if the source address comes from a network that has the same netmask or smaller as the public gateway address.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Select the Backbone realm tab at the top of the content area.

7. **IP address/Bit mask**—Enter the IP address and bitmask in the MGCP signaling interface section. The netmask should match that of the network you are masquerading in the private realm.

IP address/Bit mask

192.168.200.0/24

If the netmask is omitted, the gateway masquerading function will be disabled. Make sure that this netmask has at least as many bits as the netmask for the IPv4 address on the interface of the private realm.

8. Click **Apply**.

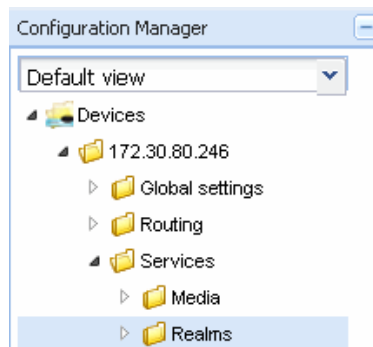
MGCP/NCS SuperNAT

The MGCP/NCS superNAT feature enables a Net-Net SBC configured to use the masquerading feature (MGCP/NCS 1:1 IPv4 Address Mapping for Gateways and Endpoints) to pass any message coming from the call agent to the gateway, even when the Net-Net SBC does not have an entry for that gateway.

If this feature is enabled and the masquerade feature is being used, the Net-Net SBC will allow the messages. If you do not enable this feature, then the Net-Net SBC only passes AUEP and DLCX messages from the call agent to the gateway if it does not have an entry for that gateway.

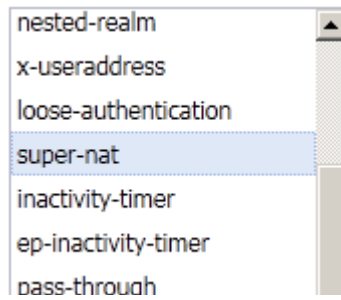
Configuring MGCP/SuperNAT

1. Expand the Configuration Manager slider > Services.

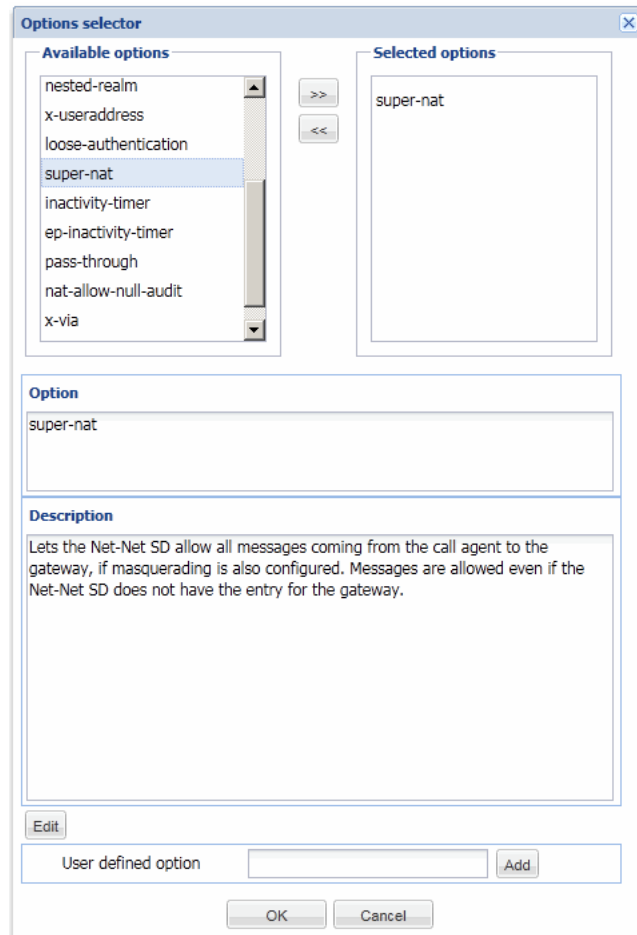


2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.

7. Click **super-nat** in the Available options list to select it.



8. Click **>>** to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.



9. Click **OK**.
10. Click **Apply**.

Call Agent Redundancy

MGCP/NCS CA redundancy can be provisioned on a Net-Net SBC to enhance its HA properties. In the event of a call agent failure, all transactions are directed to the next provisioned call agent as configured.

Call Agent Redundancy Configuration Overview

Call agent redundancy enhances the high availability of the Net-Net SBC MGCP proxy by allowing it to communicate to a secondary (or tertiary) call agent in the event that the primary call agent becomes unavailable. In the event of a CA switch over, all current and future transactions will be directed to the next provisioned CA.

CA redundancy works by incorporating DNS capability into the current MGCP/NCS implementation, and developing a ping mechanism capable of monitoring the health of a CA.

To configure CA failover, you must specify the provisioned MGCP/NCS CA as a FQDN instead of an IPv4 address. If the resolved FQDN returns more than one IPv4 address (valid "A" record), then CA redundancy will be considered activated.

Defining a Set of CAs for Redundancy

You can also manually configure individual IP addresses for the Net-Net SBC to use for provisioning redundant MGCP Call Agents (CA). The CA failover IP address parameter defines an MGCP configuration's set of redundant CAs, explicitly by IP addresses that you specify. When you do not manually configure the IP addresses, the Net-Net SBC learns of multiple CAs as returned in a DNS query.

If you configure a hostname for the public CA host and the CA redundancy parameter is enabled, the Net-Net SBC will use the IP addresses returned in the DNS response to populate its list of multiple CAs. Even though you might also configure the CA failover IP list, it will be ignored because the redundancy mechanism initiated by configuring a CA hostname takes precedence.

The order in which you configure the CA Failover IP list indicates the order in which the Net-Net SBC attempts to use each CA.

DNS Resolution for Call Agent Redundancy

If the CA redundancy feature is enabled and you do not manually define a set of CAs, the call agent is configured using its FQDN instead of its IP address. If both addressing methods are configured and DNS resolution on the FQDN is successful, address(es) returned by the DNS will take precedence.

At boot time or after an **activate config** command is issued, MGCP queries the DNS associated with the public realm network interface where the MGCP proxy exists. DNS should return at least one (and more than one if CA redundancy is desired) "A" record that identifies the IPv4 address of the primary, secondary, tertiary, and so on. The first IPv4 address returned by the DNS becomes the initial Call Agent and the others, in the order returned by the DNS, are used sequentially in failover situations.

The group of CA IP addresses returned by the DNS is known as the call agent group. The call agent group is associated with a user-defined refresh timer. The Net-Net SBC pings the members of the call agent group according to the CA Ping Interval to check that the call agents are still reachable and active.

Call Agent Failover

Call agents can fail over three ways. The two scenarios where the active call agent initiates the switch are:

- **Explicit**—The currently active CA notifies the SBC to use a different CA, also known as the notified entity. The notification method only works if the CA specified is in the form of an IPv4 address.
- **Implicit**—The currently active CA address no longer matches the source address in an incoming packet from the CA. In this case the SBC will fail over and the source address will become the new CA.

The scenario where the Net-Net SBC initiates the request is:

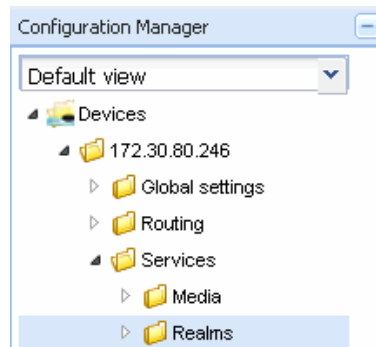
- **PING timeout**—When a CA fails to respond within the period defined in the call agent ping interval, it is considered to be out-of-service. If additional in-service call agents exist in the call agent group, the next CA will become the active call agent.

After a CA switchover, the Net-Net SBC will not initiate another switchover to return to the original state. It will only switch based on a failure. The only two situations that will force a switchover a second time are if the CA explicitly requests a switchover or if the newly active CA goes out of service.

Configuring Call Agent Redundancy

This section describes how to configure CA redundancy on your Net-Net SBC.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click the Backbone realm tab at the top of the content area.
7. **Call agent redundancy**—Select enabled from the combination box.
8. **Call agent or call agent group name**—Enter one of the following:
 - Hostname of the CA (that is identified by the required public-ca-address).
 - Hostname identifying a CA redundancy group. In order to identify an address as a CA redundancy group, it must be entered in FQDN format.
9. **IP address**—Enter the IP address for the call agent.
10. **IP port**—Enter the port number for the call agent.
11. **Ping method**—Enter the CA ping method in the Ping method field.

This parameter specifies any legal GW-originated message. After the Net-Net SBC replaces the sequence number in the given prototype message, the message is sent to the current CA. When a response is not received from the CA, it is assumed to be out-of-service. A valid prototype message could be:

NTFY [100] aaln/1@172.16.2.1

12. **Ping interval (sec)**—Enter the CA ping interval time in seconds in the Ping interval field.

The CA ping interval sets the amount of time in seconds that the Net-Net SBC waits to send a ping to the active call agent to determine if it is still healthy.

MGCP call agent

Call agent redundancy	enabled
Call agent or call agent group name	mgcp-call-agent
IP address	172.16.2.1
IP port	2727
Ping method	NTFY [100] aaln/1@172.16.2.1
Ping interval (sec)	0
Ping retries	0

13. Click **Apply**.

MGCP Call Agent Switchover

You can configure individual IP addresses for the Net-Net SBC to use for provisioning redundant MGCP Call Agents (CA). The Redundant MGCP call agents list defines an MGCP configuration's set of redundant CAs, explicitly by IP addresses that you specify.

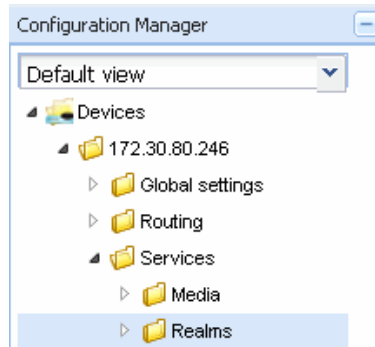
If you configure a hostname for the public CA host and the CA redundancy parameter is enabled, the Net-Net SBC will use the IP addresses returned in the DNS response to populate its list of multiple CAs. Even though you might also configure the Redundant MGCP call agents list, it will be ignored because the redundancy mechanism initiated by configuring a CA hostname takes precedence.

The order in which you configure the Redundant MGCP call agents list indicates the order in which the Net-Net SBC attempts to use each CA.

Configuring Call Agent Switchover

To configure call agent switchover:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Select the Backbone realm tab at the top of the content area.
7. Click **Add** in the Redundant MGCP call agents section. The Add Call agent dialog box appears.
8. **IP Address**—Enter the IP address for the call agent.

9. Click **Apply** to save the information and close the dialog box. The address appears in the Redundant MGCP call agents list.

Redundant MGCP call agents

Call agent

10. Repeat steps 8 through 10 to enter additional IP addresses. The order in which you configure the Redundant MGCP call agents list indicates the order in which the Net-Net SBC attempts to use each CA.
11. Click **Apply**.

Enhanced Roaming (IP Address Carrying)

The IP Address Carrying feature allows multiple active registrations for the same user at different locations. The locations are distinguished by their private IP address or, if necessary, the public address of the firewall the device is located behind. You implement this feature by including the IP addresses and port of the endpoint or firewall in two new fields included in a MGCP (or SIP) message sent from the Net-Net SBC.

For an MGCP application, the Net-Net SBC adds the following two headers in MGCP messages:

```
X-Usradd=192.168.1.10
X-Usrprt=2427
```

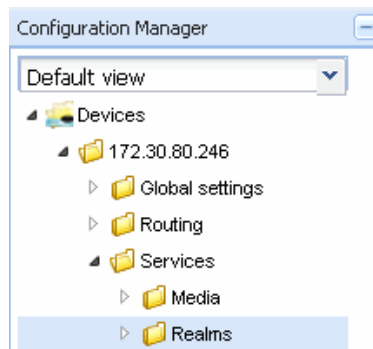
where

- X-Useradd is the MGCP equivalent of the SIP useradd field.
- X-Usrprt is the MGCP equivalent of the SIP userport field.

Configuring Enhanced Roaming

To configure enhanced roaming:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Type **x-user-info** in the User defined options field and click **Add**.

User defined option	x-user-info	Add

The x-user-add option appears in the Selected options list and a place for the option's description is created below.

8. Click **OK**.

MGCP Sans Media

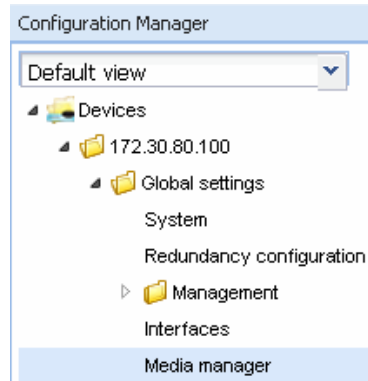
The MGCP sans media feature lets MGCP calls without media being managed by the Net-Net SBC to work. In prior releases, the Net-Net SBC always managed the

media for MGCP, you could not configure it to do otherwise. With this release, you can set the media manager's state parameter to disabled to let the Net-Net SBC pass SDP to the endpoints without managing the media.

Configuring MGCP Sans Media

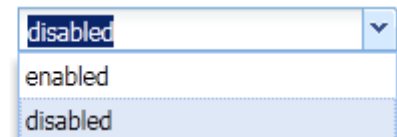
To configure MGCP sans media:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. **State**—Choose **disabled** from the combination box.

State



5. Click **Apply**.

MGCP/NCS X-Keepalives

Some VoIP phones have a *keep-alive* option to support intermediate NATs. This option involves the periodic transmission of RSIPs that include an X-Keepalive parameter.

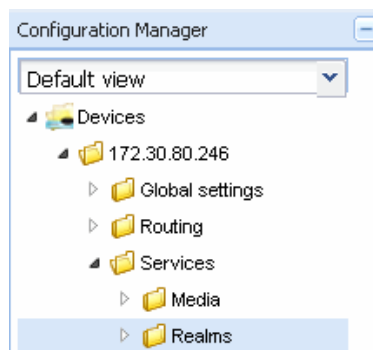
The Net-Net SBC's default method of performing Hosted NAT Traversal (HNT) in MGCP/NCS environments is to send periodic AUEP messages to each endpoint. If it is sent prior to the NAT binding time-to-live (TTL) expiry time, this traffic refreshes the binding on the intermediary NAT device. This HNT technique is referred to as a "push"; that is, the hosted device pushes keepalive messages to each endpoint.

When the Net-Net SBC receives an RSIP with an X-Keepalive header present, it checks to see if that endpoint is known.

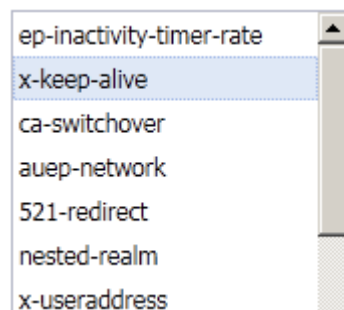
- If it is known, the Net-Net SBC responds with a 200 OK directly and suppresses the AUEP keepalives that are generated.
- If it is not known, the Net-Net SBC replaces the X-Keepalive header with Restart and forwards it to the call agent.

Configuring X-Keepalives

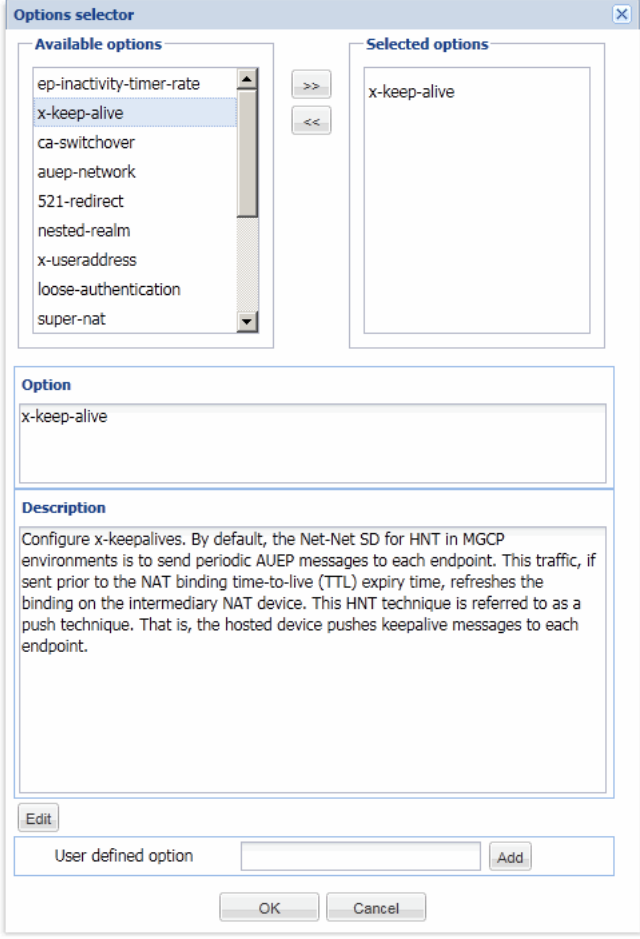
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **x-keep-alive** in the Available options list to select it.



8. Click  to move the option to the Selected options list. The option name appears along with a description.



The **Options selector** dialog box is shown. It has a title bar with a close button. The dialog is divided into several sections:

- Available options:** A list box containing the following options: ep-inactivity-timer-rate, x-keep-alive (highlighted), ca-switchover, auep-network, 521-redirect, nested-realm, x-useraddress, loose-authentication, and super-nat. To the right of this list are two buttons: a right arrow (>>) and a left arrow (<<).
- Selected options:** A list box containing the option x-keep-alive.
- Option:** A text field containing the text "x-keep-alive".
- Description:** A text area containing the following text: "Configure x-keepalives. By default, the Net-Net SD for HNT in MGCP environments is to send periodic AUEP messages to each endpoint. This traffic, if sent prior to the NAT binding time-to-live (TTL) expiry time, refreshes the binding on the intermediary NAT device. This HNT technique is referred to as a push technique. That is, the hosted device pushes keepalive messages to each endpoint."
- Edit:** A button located below the description.
- User defined option:** A section with a text input field and an **Add** button.
- Buttons:** At the bottom of the dialog are **OK** and **Cancel** buttons.

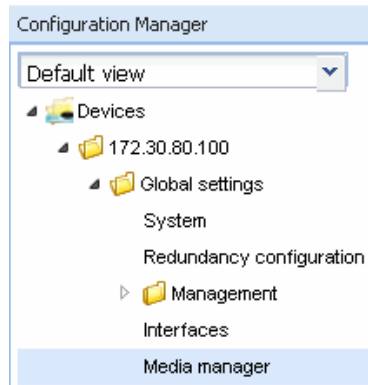
9. Click **OK**.
10. Click **Apply**.

MGCP Congestion Control

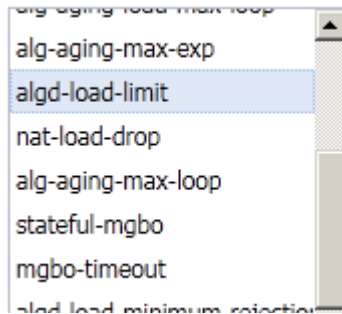
You can limit the amount of CPU the MGCP task consumes.

Configuring MGCP Congestion Control

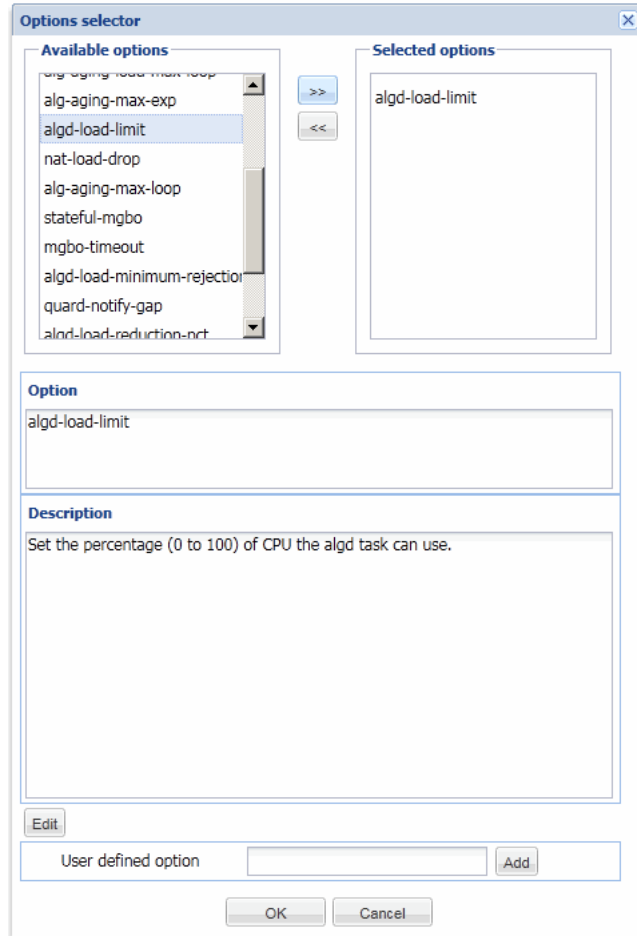
1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **algd-load-limit** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.



7. Click **OK**.
8. Click **Apply**.

Graceful Stateful Backoff

The MGCP congestion control helps customers handle large call events in an oversubscribed environment. When enabled, the Net-Net SBC sends a system busy message back to the call agent for new calls when system resources have been exhausted. Overload conditions are determined by CPU utilization. When a utilization threshold is exceeded, the Net-Net SBC issues an alarm and a corresponding trap.

Because some users have experienced a “ghost ring”(indefinite ringing on the side of the caller and dead air on the side of the called person when they answer), a new feature called MGCP graceful stateful backoff has been developed. In the previous MGCP congestion control implementation, the Net-Net SBC did not distinguish between NTFY messages from the calling party and the called party.

How It Works

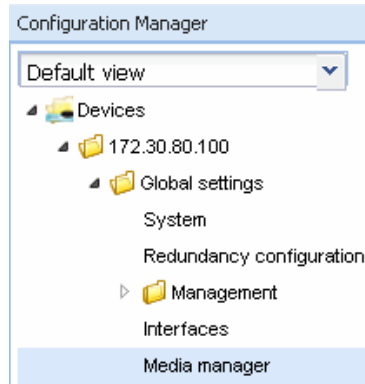
When enabled, the Net-Net SBC creates a list of calls rejected due to overload. When a call is rejected, the Net-Net SBC checks both the request identifier and endpoint IP addresses to see if that particular call was rejected due to overload. If a call has been rejected for this reason, the call is passed and no ghost ring occurs.

Configuring Graceful Stateful Backoff

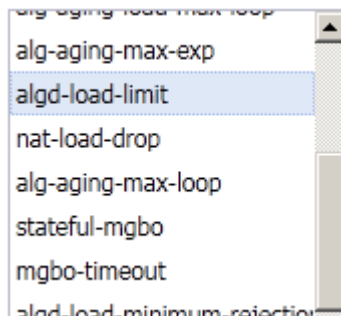
You configure the **algd-load-limit** option to enable MGCP graceful stateful backoff. The percentage you enter here specifies the CPU utilization threshold that triggers the rejection of off-hook Notify (NTFY) messages.

To configure MGCP graceful stateful backoff:

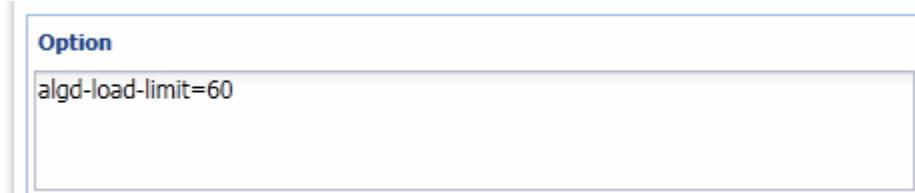
1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **algd-load-limit** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
7. Edit the textbox in the Description section to indicate the percentage of CPU utilization to use as the threshold.



8. Click **OK**.
9. Click **Apply**.

MGCP Endpoint Aging

When an MGCP gateway registers through the Net-Net SBC, memory is allocated to hold state information associated with each active endpoint on the gateway. In a number of circumstances this memory is never released. An endpoint aging mechanism has been added to the Net-Net SBC to remove state information that is no longer needed and free up associated memory.

How It Works

The Net-Net SBC maintains a per-endpoint timer to track when traffic was last received from the gateway. If the timer expires, the Net-Net SBC deletes the endpoint and frees its resources. If all endpoints associated with a gateway are deleted, then the Net-Net SBC removes the gateway entry, too.

Any traffic received from an endpoint resets the activity timer for that endpoint.

Dynamic Reconfiguration

This feature is RTC-supported. The Net-Net SBC applies newly configured values when an inactivity timer expires or when it creates a new endpoint entry.

However, when an activity timer is currently set to 0 or being set to 0, the Net-Net SBC sets all inactivity timers to the new value. However, the Net-Net SBC staggers the inactivity timers because it is undesirable to have a large number of endpoints deleted at the same time.

Considerations for HA

Inactivity timers are not replicated across the Net-Net SBCs in an HA node. If there is a switchover, the inactivity timers for all endpoints are initialized for the amount of time you set. However, the Net-Net SBC staggers the inactivity timers because it is undesirable to have a large number of endpoints deleted at the same time.

Deletion Smoothing

Built into this feature is a smoothing mechanism that, when necessary, staggers the deletion of endpoints to provide continuity in system resource use.

When you enable this feature, you set two parameters. One is the time value for the number of seconds after which an endpoint is considered inactive and is deleted. A second time value defines how many milliseconds to add to an endpoint's inactivity timer in order to stagger its time from the previous timer.

Note the following:

- If Endpoint 1 and Endpoint 2 send messages outside the window defined in the inactivity time rate, then the Net-Net SBC does not stagger deletions.
- If the MGCP endpoint sends a message (any kind of MGCP message) within the allowable time, then the Net-Net SBC resets the timer for that endpoint.

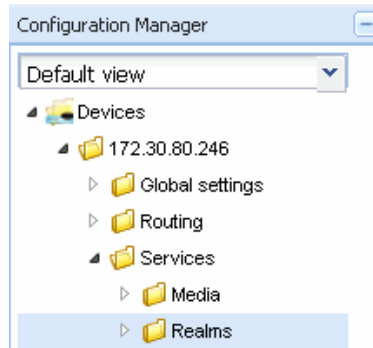
Configuring Endpoint Aging

You enable MGCP endpoint aging on the Net-Net SBC by setting two options in the MGCP configuration:

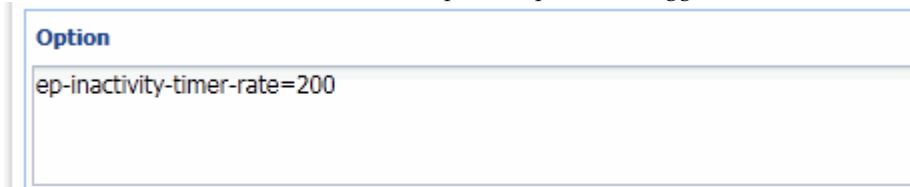
- **ep-inactivity-timer=X**—Sets the amount of time in seconds before an MGCP endpoint expires; to stagger deletions, the **ep-inactivity-timer-rate** value can be added to the inactivity timer if multiple endpoints send messages within the inactivity timer rate window
- **ep-inactivity-timer-rate=X**—Defines a minimum gap in milliseconds between timers so that multiple expiring timers do not cause a CPU utilization spike; if you do not set a time, the Net-Net SBC uses a 100 millisecond default

To enable MGCP endpoint aging:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **ep-inactivity-timer-rate** in the Available options list to select it.
8. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
9. **Description**—Edit the textbox in the Description section to set the minimum gap in milliseconds between timers.
10. Repeat steps 8 and 9 to select to move the **ep-inactivity-timer** option to the Selected options list.
11. **Description**—Edit the textbox in the Descriptions section to set the amount of time in seconds before an MGCP endpoint expires; to stagger deletions.



12. Click **OK**.
13. Click **Apply**.

Endpoint Aging Optimization

With MGCP endpoint aging, the endpoint clean-up can consume a great deal of Net-Net SBC resources and so slow call processing. To prevent slow call processing, you can make the endpoint timers controlled (rather than actual). This way, the Net-Net SBC controls the rate at which endpoints age and how many endpoints to age, and keeps CPU load to a minimum. If the CPU load exceeds the threshold, controlled timers automatically reduce the rate of MGCP endpoint aging.

How It Works

You configure media manager options to optimize Net-Net SBC MGCP endpoint aging. For the options to apply, you must have enabled CPU limiting and inactivity timers.

The options determine how many endpoints are checked during the two-second controlled timer period, and also how many endpoints the Net-Net SBC is allowed to age during that loop. They also set the CPU threshold and define how many endpoints are checked and how many can be aged when the threshold is exceeded.

The options you can configure are defined in the following table.

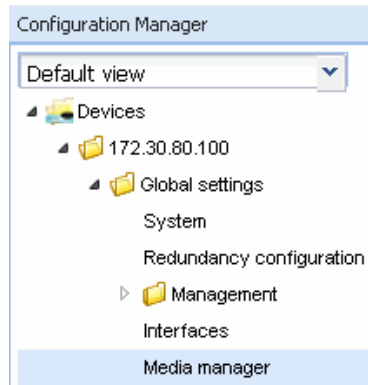
Option Name	Default Value	Description
alg-aging-max-loop	100 (integer)	Maximum number of endpoints examined for every controlled timer tick (two seconds) to determine aging
alg-again-max-exp	15 (integer)	Maximum number of endpoints the Net-Net SBC will age per loop, limits a flood of endpoints from being aged at the same time
alg-aging-cpu-limit	30 (integer)	Threshold for percentage of CPU; if CPU utilization exceeds this threshold, alg-aging-load-lax-loop and alg-aging-load-max-exp are used
alg-aging-load-lax-loop	50 (integer)	When CPU threshold has been exceeded, maximum number of endpoints examined for every controlled timer tick (two seconds) to determine aging
alg-aging-load-max-exp	3 (integer)	When CPU threshold has been exceeded, acceptable maximum number of endpoints the Net-Net SBC will age per loop, limits a flood of endpoints from being aged at the same time

Configuring Endpoint Aging Optimization

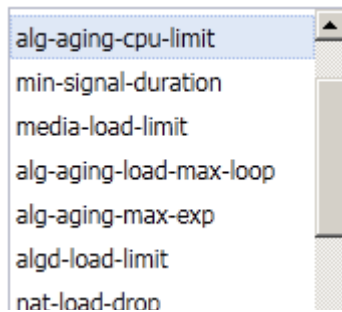
Acme Packet recommends that you use the default values for this feature. If you change the defaults, do so with caution because faulty values can degrade system performance.

To configure MGCP endpoint aging optimization:

1. Expand the Configuration Manager slider > Global Settings.

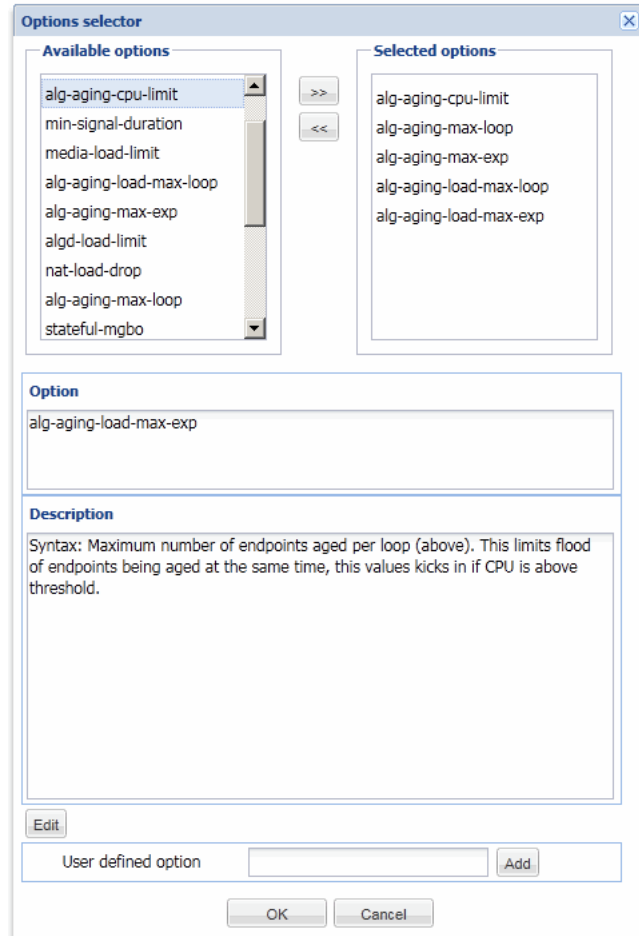


2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Click **Options**. The Options dialog box appears.
5. Click **alg-aging-cpu-limit** in the Available options list to select it.



6. Click  to move the option to the Selected options list.

7. Repeat steps 5 and 6 to move all MGCP endpoint aging options to the Selected options list.



8. **alg-aging-cpu-limit**—Edit the textbox to enter the CPU percentage you want to use as a threshold.

alg-aging-cpu-limit=30

If CPU usage exceeds the threshold, the following options are used instead:

- **alg-aging-load-max-loop**—Edit the option textbox to enter the number of endpoints you want examined for every controlled timer tick (two seconds) to determine aging

alg-aging-load-max-loop=75

- **alg-aging-load-max-exp**—Edit the options textbox to enter the acceptable maximum number of endpoints the Net-Net SBC will age per loop, which limits a flood of endpoints from being aged at the same time

alg-aging-load-max-exp=5

9. **alg-aging-max-loop**—Edit the textbox to enter the maximum number of endpoints examined for every controlled timer tick (two seconds) to determine aging.

alg-aging-max-loop=150

10. **alg-aging-max-exp**—Edit the textbox to enter the maximum number of endpoints the Net-Net SBC will age per loop, which limits a flood of endpoints from being aged at the same time.

alg-aging-max-exp=15

11. Click **OK** to save your edits and close the options window.
12. Click **Apply**.

Endpoint Removal

You can configure your Net-Net SBC for MGCP configurable endpoint removal, a feature that enables dynamic removal of MGCP endpoints and related session information upon receipt of configured 5xx series permanent error code(s).

In MGCP, a gateway notifies a call agent that a group of endpoints managed by the gateway is going on or out of service by sending a RestartInProgress (RSIP) message. The call agent's response to the RSIP contains a code reflecting the success or failure of the request:

- **Success**—Indicated by the return code 200 - transaction executed; restart was completed successfully, and the NotifiedEntity returned is the new notified entity for the endpoint
- **Failure**—Either indicated by 4xx (transient error) or 5xx (permanent error) return codes; only the 5xx codes are relevant to this feature; restart fail to complete successfully; if there is a NotifiedEntity returned in the response, this new notified entity must be include when the restart is attempted again

For more information about these classes of codes and about individual codes, see RFC 3435, "Media Gateway Control Protocol (MGCP) Version 1.0."

In prior releases, the Net-Net SBC created an MGCP session when it received the RSIP message. This behavior could leave the Net-Net SBC vulnerable to a DoS attack if the attacker were to willfully misrepresent thousands of endpoints, eventually exhausting the Net-Net SBC's memory capacity.

How It Works

You can set the 5xx return codes—or series of 5xx codes—that trigger endpoint removal. Because the request creates the endpoint and reserves associated Net-Net SBC resources for it, RSIP are the basis for removal; the Net-Net SBC disregards other requests if the endpoint is not registered first.

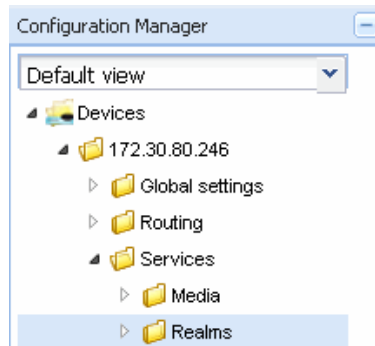
The default value for the parameter is a set of ranges: **500-509, 511-519, 522-599**. This set of return codes excludes numbers 510 (unspecified protocol error) and 520 (endpoint is restarting and the transaction could not be completed) because the errors they reflect are temporary, for which sessions should be established.

You can include return code 521 (redirection) without creating conflict with the existing feature, which removes the MGCP session when a 520 is received. An empty-string entry disables the feature.

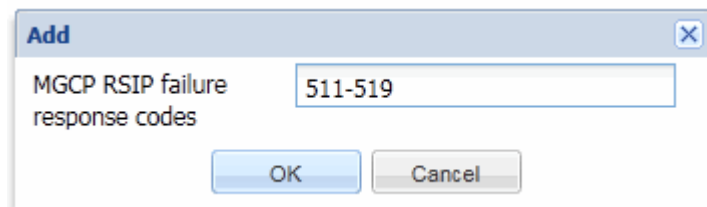
Configuring MGCP Endpoint Removal

To configure MGCP endpoint removal:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Click the Backbone realm tab.
6. Scroll to the MGCP RSIP failure response codes table.
7. Click **Add**. The Add MGCP RSIP failure response codes dialog box appears.
8. **MGCP RSIP failure response codes**—Enter the range of 5xx return codes that you want to trigger MGCP endpoint removal or that you do not want to succeed in creating an MGCP session on the Net-Net SBC.



9. Click **OK** to close the dialog box. The response code range appears in the MGCP RSIP failure response codes list.
10. Repeat steps 8 through 10 to add additional response codes.
11. Click **Apply**.

MGCP Restricted Latching

The Net-Net SBC now supports restricted media latching for MGCP. Restricted latching offers security from rogue RTP packets by isolating the valid sources of RTP flows to well-known, signaled addresses.

When restricted media latching is enabled, the Net-Net SBC latches only to media from known source IP addresses in order to learn and latch onto the dynamic UDP port number. The IP address's origin can be either the SDP information or the SIP message's Layer 3 IP address, depending on your configuration.

The Net-Net SBC restricts latching of RTP/RTCP media for all calls within a realm. It latches to media based on one of the following:

- **SDP**—The IP address and address range based on the received SDP c= connect address line in the offer and answer
- **Layer 3**—The IP address and the address range based on the received L3 IP address of the offer or answer.

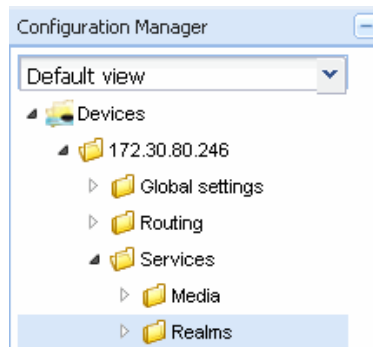
For more information about restricted latching (and its relationship to symmetric latching), refer to the *Realms and Nested Realms* chapter of the *Net-Net Configuration Guide*. This feature does not have any impact on the **latching** parameter in the media manager configuration.

Configuring Restricted Latching

This section explains how to configure restricted latching on the Net-Net SBC. These are the same parameters previously used for SIP only, but now they also support this feature for MGCP.

To configure restricted latching:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Scroll to Restricted latching.
4. **Restricted latching mode**—Choose the latching mode from the combination box in the Restricted latching section.
 - **none**—no latching used
 - **sdp**—use the address provided in the SDP for latching
 - **peer-ip**—use the layer 3 signaling address for latching
5. **Length of bit map for sourced latched address**—Enter the number of address bits you want used for the source latched address. If set to 32, the complete IP

address is matched. This field will be used only if the restricted latching mode is `sdp` or `peer-ip`.

Restricted latching

Restricted latching mode	peer-ip
Length of bit map for source latched address	32

- Click **Apply**.

MGCP Port Mapping

The Net-Net SBC supports gateway masquerading, which gives an external gateway a unique address representation on the core side using a one-to-one mapping of IP addresses. In addition, the Net-Net SBC supports MGCP port mapping, which allows for mapping many IP addresses to one IP address on the core gateway side. This mapping uses unique and dynamically allocated ports for each external gateway. You can configure the range from which the port numbers are drawn.

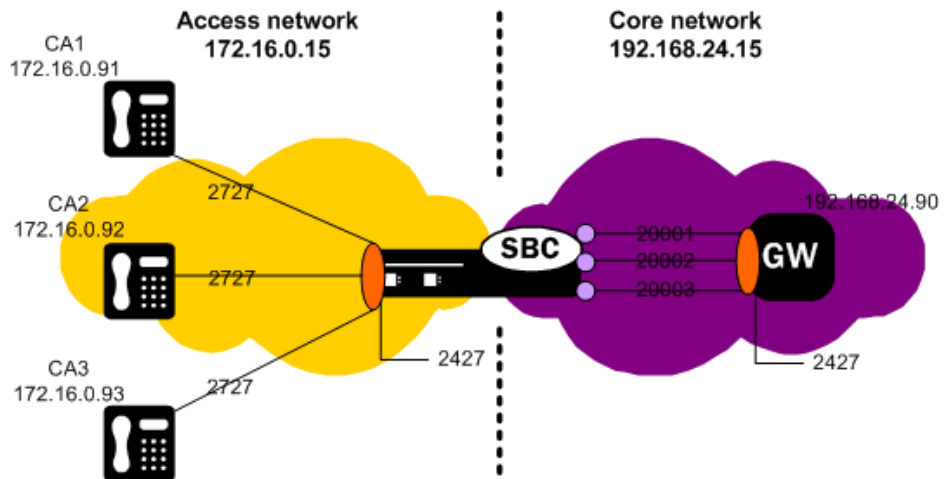
MGCP port mapping eliminates the need to reserve a large number of IP addresses to use for one-to-one mapping on the Net-Net SBC, and from having to provision a large number of static routes in core network router. This is because when traffic from an endpoint in the public Internet traverses the Net-Net SBC, the Net-Net SBC designates a specific port number to represent the endpoint. The core gateway IP address on the Net-Net SBC remains the same; only the port number changes.

How It Works

For admission control, certain service providers require a unique signaling transport address (IP address and UDP port combination) for each endpoint. However, when deployed in such a network, the Net-Net SBC sits between endpoints and the gateway and its IP is perceived address as the IP address of the individual endpoints. To give endpoints unique transport addresses, the Net-Net SBC allocates a port number for each.

When it receives a session-initiating MGCP message from the access side, the Net-Net SBC refers to a pool of ports you have configured to allocate a unique signaling port. Then, when it forwards an RSIP, the Net-Net SBC replaces the source address and port information with the allocated signaling port address.

The following diagram illustrates how the MGCP port mapping feature works for an access deployment where the MGCP configuration's port mapping feature is enabled.



In this figure, CA1, CA2, and CA3 are call agents in the access network, for which the IP address and port value is 172.16.0.15:2427. The configuration for this sample would also show a public gateway address (the **IP address/Bit mask** parameter) of 192.168.24.15. The gateway resides at 192.168.24.90:2427. The Net-Net SBC allocates ports 20001 through 20003 for the three calls agents on the access side. The gateway perceives each as having a unique signaling address, rather than all having the signaling address of the Net-Net SBC itself. For example, if CA1 were to send an RSIP with a source IP address and port of 172.16.0.91:2727, the Net-Net SBC replaces that information with 192.168.24.15:20001 when it forwards the RSIP to the core network. Likewise, all MGCP requests addressed to the allocated signaling transport address are translated and forwarded with the associated MGCP session.

Availability of Ports in the Pool

Signaling ports are returned to the pool for use with new MGCP sessions when the MGCP session for which they were being used is removed. Removal might occur when a session times out, or when it is explicitly removed with an RSIP request. The newly available port is returned at the bottom of the list of available ports, resulting in a least-used allocation method for signaling port selection.

If there are no ports available for a new registration, the request uses the IP address and port configured for—in this order—the **EMS IP address/Bit mask** and the **IP port transmit** parameter values.

Configuring MGCP Port Mapping

To enable this feature, you configure start and end values for the range of port numbers you want the Net-Net SBC to allocate to individual endpoints. Your configuration must follow these rules or your port range will be invalid. Invalid port ranges prevent this feature from working properly.

The range of ports you enter:

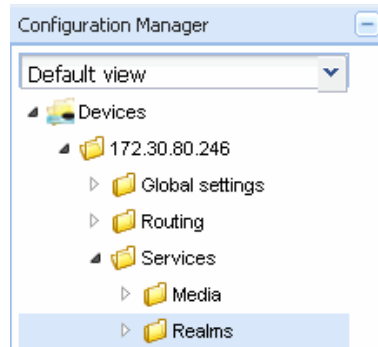
- Must not overlap with configured IP ports (as defined in the Backbone realm tab/MGCP call agent) because it might be used for signaling messages that are not associated with MGCP sessions.

- Must not overlap the IP address/Bit mask (as defined in the Backbone realm tab/MGCP signaling interface) configuration using the same IP address as the public gateway address.

Overlap prevents the Net-Net SBC from internally processing traffic as it should. The ability to define a range of ports increases the potential for overlap.

To define a port range that enables MGCP port mapping:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click the Backbone realm tab at the top of the content area.
7. Scroll to Gateway port mapping.
8. **Start port**—Enter the port number marking the beginning of the range of ports you want to use for MGCP port mapping. The valid range is 1025 through 65535. If you leave this parameter set to its default of 0 the feature is disabled.
9. **End port**—Enter the port number marking the end of the range of ports you want to use for MGCP port mapping. The valid range is 1025 through 65535. If you leave this parameter set to its default of 0 the feature is disabled. The **End port** value you set must be greater than the **Start port** value.

Start port	1025
End port	65535

10. Click **Apply**.

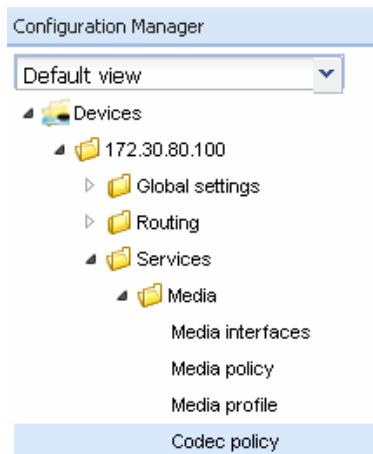
Reordering and Stripping Multiple Codecs for MGCP

This feature builds on the Net-Net SBC's pre-existing abilities to route by codec and reorder one codec in an SDP offer. This functionality allows the reordering of multiple codecs, the removal of specific codecs, and also the removal of codec types in SDP offers. You can enable the Net-Net SBC to perform these operations on SDP offers by configuring codec policies. Codec policies are sets of rules that specify the manipulations to be performed on SDP offers. They are applied on an ingress basis using the realm configurations.

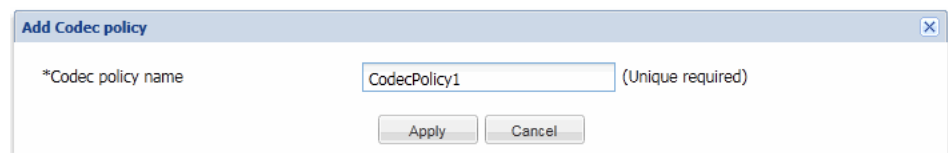
Creating a Codec Policy

To create a codec policy:

1. Expand the Configuration Manager slider > Services > Media.

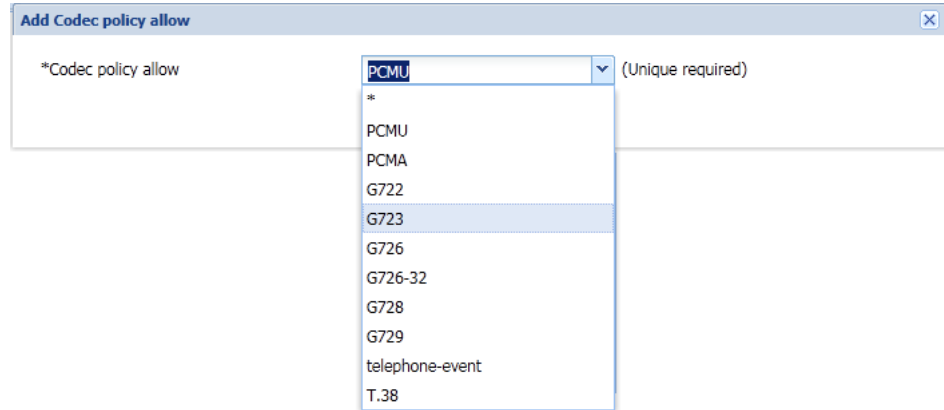


2. Click Codec policy. The Codec policy table opens in the content area.
3. Select a Codec policy and click **Edit**.
4. Click **Add**. The Add Codec policy dialog box appears.
5. **Codec policy name**—Enter the unique name for the codec policy. You refer to this codec policy name to apply it to realms or session agents. This parameter is required.



6. Click **Apply** to save the name and close the dialog box. The Codec policy name appears in the table.
7. Click the Codec in the table to select it and click **Edit**. The Edit Codec policy dialog box appears.
8. For Codec policy allow, click **Add**. The Add Codec policy allow dialog box appears.

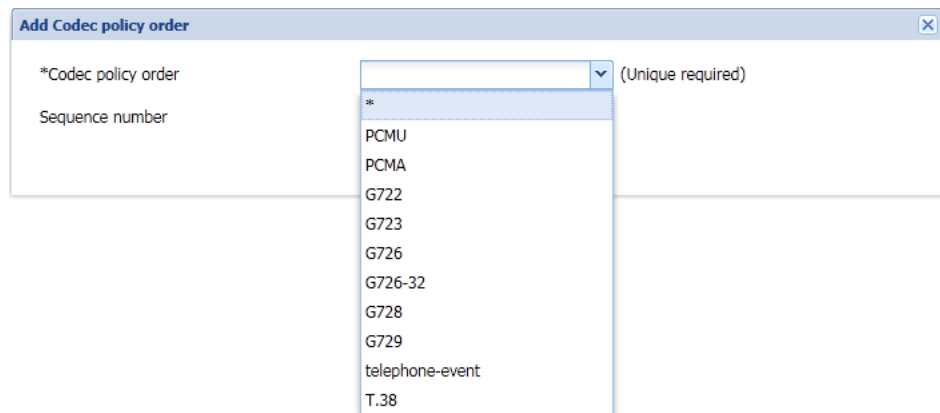
9. **Codec policy allow**—Choose the media format type (codec) to allow for this codec policy from the combination box.



You can use the asterisk (*) as a wildcard, the force attribute, or the no attribute so that the allow list you enter directly reflects your configuration needs. For more information, refer to the Manipulation Modes section.

Note: The codecs that you enter here must have corresponding media profile configurations.

10. Click **Apply** to close the dialog box. The codec appears in the Codec policy allow list.
11. For Codec policy order, click **Add**. The Add Codec policy order dialog box appears.
12. **Codec policy order**—Choose the codec from the combination box.



The order in which you list the codecs is the order in which they appear in the outgoing SDP offer. You can use the asterisk (*) as a wildcard in different positions of the order to directly reflect your configuration needs. For more information, refer to the Manipulation Modes section.

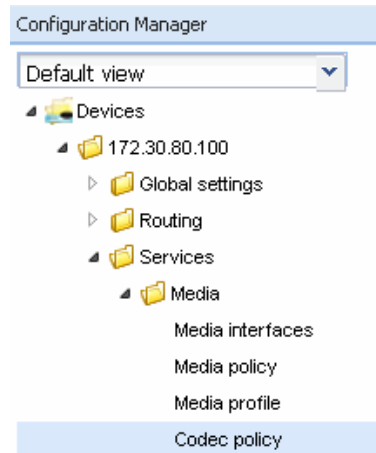
Note: The codecs that you enter here must have corresponding media profile configurations.

13. Click **Apply** to close the dialog box. The codec appears in the Codec policy order list.
14. Click **Apply**.

Reordering Codecs

To reorder codecs:

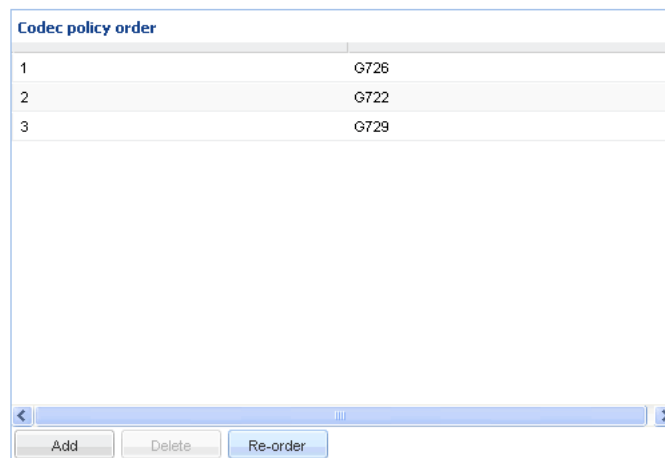
1. Expand the Configuration Manager slider > Services > Media.



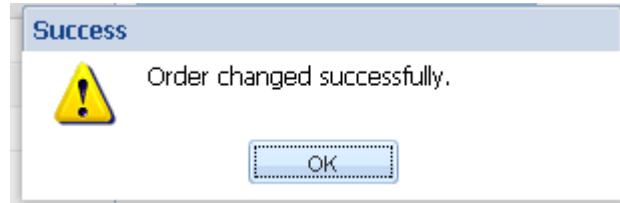
2. Click Codec policy. The Codec policy table opens in the content area.
3. Select a Codec policy and click **Edit**.
4. Click a codec policy in the table and click **Edit**.
5. Scroll to Codec policy reorder.
6. Select a codec policy by clicking it, and drag the policy to new slot.

1	PCMU
2	PCMA
3	*
4	G723

7. Click Reorder.



8. A Success dialog box appears. Click **OK**. The new order is displayed in the Codec policy order list.



9. Repeat steps 6-8 to re-order additional codecs.
10. Click **Apply**.

Deleting Codecs

To delete codecs from a policy:

1. Click a codec policy in the table and click **Edit**. The Edit Codec policy dialog box appears.
2. **Codec policy allow**—Click the codec policy name in the list and click **Delete**. A confirmation message appears.
3. Click **OK** to clear the message. The codec policy is removed from the Codec policy allow list.
4. If the codec policy is also in the **Codec policy reorder** list, follow the same actions to remove it.
5. Click **Apply**.

MGCP Configurable CPU Sample Rate

The MGCP configurable sampling rate prevents CPU resource exhaustion by allowing you to set the rate at which you want CPU measured. To perform MGCP congestion control, the Net-Net SBC periodically measures CPU utilization. In prior releases, the preset for this measurement is ten seconds during normal usage states and five seconds during overload conditions.

The media control options described here let you set the measurement interval to a time between one and ten seconds. For overload conditions, this measurement interval is set to five seconds if the value you set is greater than five. You can also set parameters around rejection percentages and standby time.

How It Works

The CPU sampling rate options let you set the measurement interval for CPU sampling, rejections percentages, and the state of the Net-Net SBC. Some of these value are used in Net-Net SBC calculations that guard against sudden drops in rejection rate.

- **algd-load-measurement-interval**—This value controls how often the Net-Net SBC's MGCP task checks the CPU utilization of application tasks. A small interval makes the task respond more quickly to changes in CPU utilization, but adds overhead to the CPU utilization of the task individually and to the overall CPU utilization of application tasks.

The valid range for this option is one to ten seconds. During an overload condition, this rate automatically defaults to five seconds if the value you set is greater than five seconds. All subsequent options described in this section depend on this option.

- **algd-load-average-number-periods**—When you define this option, the Net-Net SBC's MGCP task also incorporates the average CPU utilization into its decision to enter congestion state (and start sending 403 messages to new NTFY off-hook commands). The Net-Net SBC computes average CPU utilization over the period of time equal to **algd-load-measurement-interval** multiplied by the **algd-load-average-number-periods** value. The MGCP task enters congestion state when both the current CPU utilization and the average CPU utilization exceed the **algd-load-limit** value.
- **algd-load-reduction-pct**—When you define this option, the Net-Net SBC uses its value to compute a new rejection rate if there is a decrease in CPU utilization during congestion state and standby state. The new rejection rate equals 100 minus the **algd-load-reduction-pct** value, as a percentage of the previous rejection rate.

For example, if the previous rejection rate is 10% and the `algd-load-reduction-pct` value is 10%, then the new rejection rate is 90% of 10%, which is 9%.

The valid range is 1 to 100, with a default of 0.

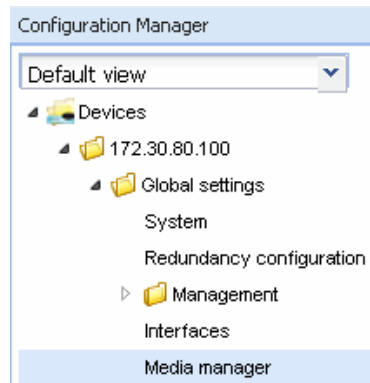
- **`algd-load-minimum-rejection-rate`**—When you define the `algd-load-reduction-pct`, this options specifies the minimum rejection rate when computed using the `algd-load-reduction-pct` value. If a new rejection rate is less than the value set using this option, then the Net-Net SBC sets the new rejection rate to 0. The default is 5.
- **`algd-load-standby-period`**—Time in seconds that the Net-Net SBC stays in a standby state after CPU utilization has receded below the load limit and before it goes back to the normal state. If overload conditions return, the Net-Net SBC returns to the overload state.

The valid range is 1 to 60, with a default of 60.

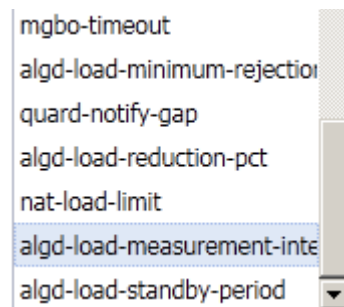
Configuring MGCP Sampling Rate

To set the appropriate options for configurable CPU measurement intervals:

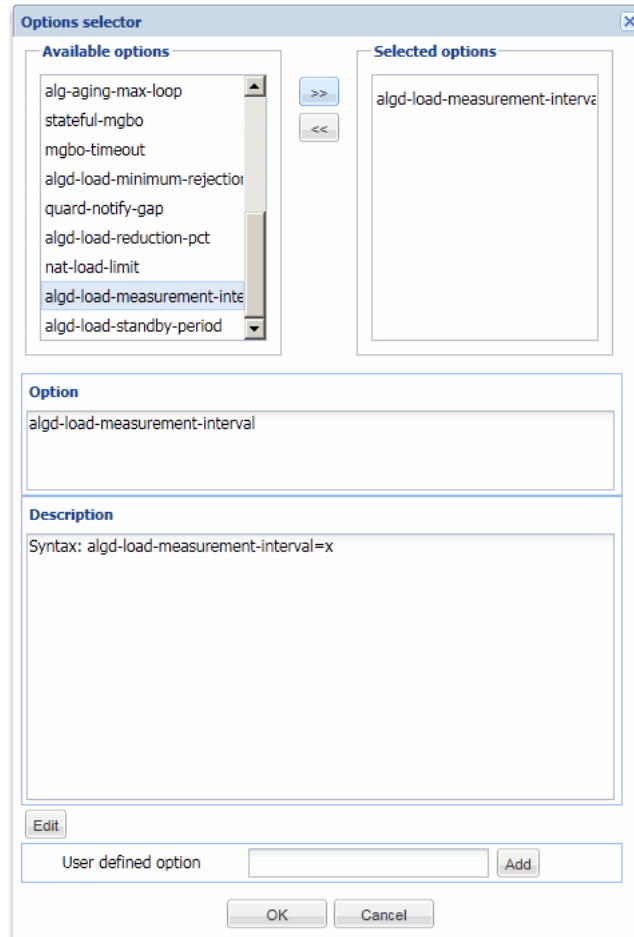
1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Scroll to the Options section and click **Options**. The Options dialog box appears.
5. Click **`algd-load-measurement-interval`** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has two main sections: **Available options** and **Selected options**. In the **Available options** list, `algd-load-measurement-interval` is selected. The **Selected options** list is currently empty. Below these lists, the **Option** field displays `algd-load-measurement-interval` and the **Description** field shows the syntax `Syntax: algd-load-measurement-interval=x`. At the bottom, there is an **Edit** button, a **User defined option** text box, an **Add** button, and **OK** and **Cancel** buttons.

7. **algd-load-measurement-interval**—Edit the textbox to append the number of seconds for the interval at which the Net-Net SBC's MGCP task checks the CPU utilization of application tasks. For example:
- `algd-load-measurement-interval=5`
8. Repeat steps 4 through 6 to add and edit the other options for the other CPU sampling rate options.
9. Click **OK** to save your edits and close the Options dialog box.
10. Click **Apply**.

MGCP: NAT Traversal without Audit Interval

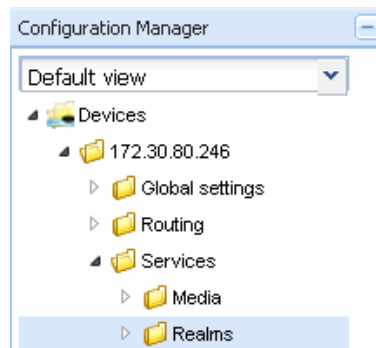
In many MGCP configurations, you use NAT traversal by setting the **NAT traversal** parameter to enabled and setting the **Audit interval** parameter to a non-zero value. But by setting the **nat-allow-null-audit** option in the MGCP configuration, you enable your system to use NAT traversal without setting the **Audit interval**.

Configuring MGCP to Allow NAT Traversal Without Audits

This section shows you how to configure your system to use NAT traversal for MGCP traffic without using the audit interval.

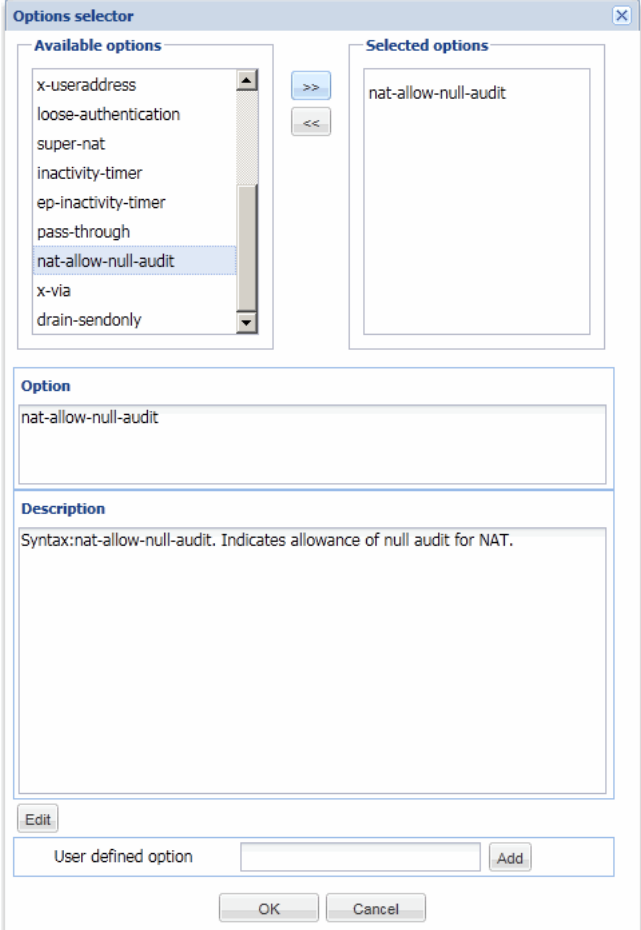
To set the nat-allow-null-audit option:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select an MGCP realm from the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select an mgcpConfig service from the table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **nat-allow-null-audit** in the Available options list to select it.

8. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.



The **Options selector** dialog box is shown. It has a title bar with a close button. The main area is divided into two panes: **Available options** on the left and **Selected options** on the right. The **Available options** pane contains a list box with the following items: x-useraddress, loose-authentication, super-nat, inactivity-timer, ep-inactivity-timer, pass-through, nat-allow-null-audit (highlighted), x-via, and drain-sendonly. Between the panes are two buttons: a right arrow (>>) and a left arrow (<<). The **Selected options** pane contains a list box with the item: nat-allow-null-audit. Below the panes, there are two sections: **Option** and **Description**. The **Option** section contains a text box with the value "nat-allow-null-audit". The **Description** section contains a text box with the value "Syntax:nat-allow-null-audit. Indicates allowance of null audit for NAT." Below these sections is an **Edit** button. At the bottom, there is a **User defined option** section with a text box and an **Add** button. At the very bottom are **OK** and **Cancel** buttons.

9. Click **OK**.
10. Click **Apply**.

9 DNS Application Layer Gateway Service

Introduction

The Net-Net SBC's DNS Application Layer Gateway (ALG) feature provides an application layer gateway for DNS transactions on the Net-Net SBC. With DNS ALG service configured, the Net-Net SBC can support the appearance of multiple DNS servers on one side and a single DNS client on the other.

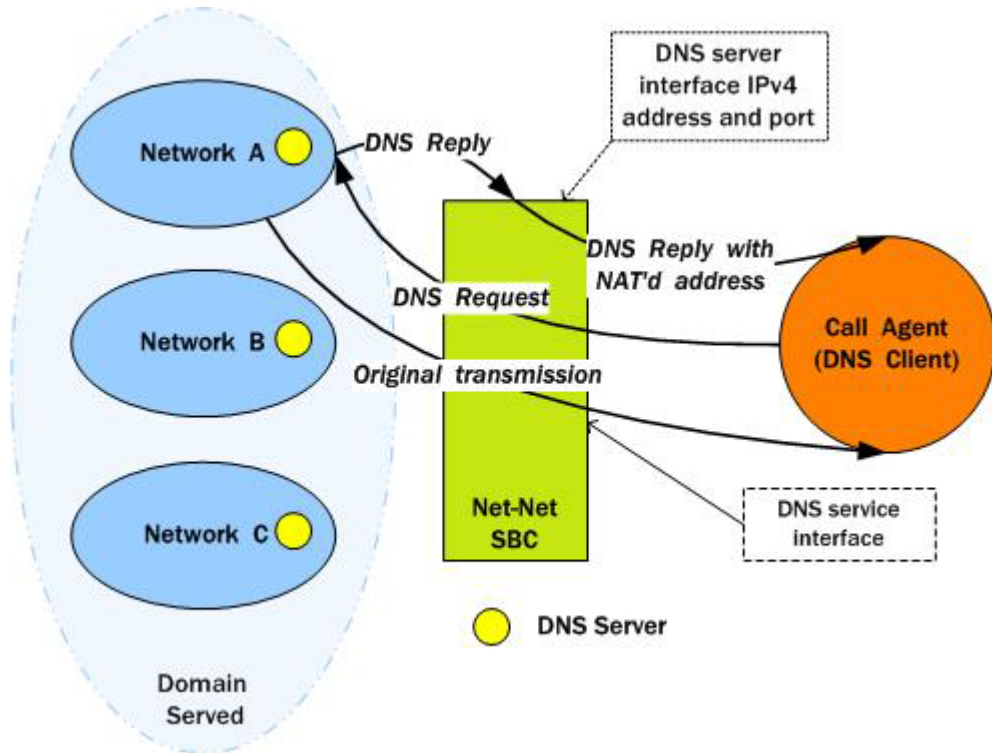
Overview

DNS ALG service provides an application layer gateway for use with DNS clients. DNS ALG service allows a client to access multiple DNS servers in different networks and provides routing to/from those servers. It also supports flexible address translation of the DNS query/response packets. These functions allow the DNS client to query many different domains from a single DNS server instance on the client side of the network.

The Net-Net SBC's DNS ALG service is commonly used when a DNS client (such as a call agent) needs to authenticate users. In this case, the DNS client that received a message from a certain network would need to authenticate the endpoint in a remote network. Since the DNS client and the sender of the message are on different networks, the Net-Net SBC acts as an intermediary by interoperating with both.

In the following diagram, the DNS client has received a message from an endpoint in Network A. Since the DNS client is in a different realm, however, the DNS client receives the message after the Net-Net SBC has performed address translation. Then the DNS client initiates a DNS query on the translated address. The Net-Net SBC forwards the DNS request to the DNS server in Network A, using the domain suffix to find the appropriate server. Network A's DNS server returns a response containing its IPv4 address, and then the Net-Net SBC takes that reply and performs

a NAT on the private address. The private address is turned into a public one that the DNS client can use to authenticate the endpoint.



Configuring DNS ALG Service

This section tells you how to access and set the values you need depending on the configuration mechanism you choose. It also provides sample configurations for your reference.

Configuring DNS ALG service requires that you carry out two main procedures:

- Setting the name, realm, and DNS service IPv4 interfaces
- Setting the appropriate parameters for DNS servers to use in other realms

Before You Configure

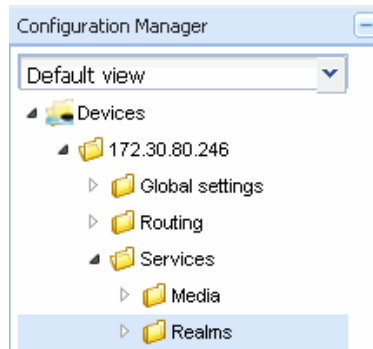
Before you begin to configure DNS ALG service on the Net-Net SBC, complete the following steps.

1. Configure the client realm that you are going to use in the main DNS ALG profile and note its name to use in this chapter's configuration process.
2. Configure the server realm that contains the DNS servers and note its name to use in this chapter's configuration process.
3. Determine the domain suffixes for the network where the DNS servers are located so that you can enter them in the domain suffix parameter.
4. Devise the NAT scheme that you want to use when the DNS reply transits the Net-Net SBC.

Configuring DNS ALG

This section describes how to configure DNS ALG settings on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. Click the DNS-ALG tab at the top of the content area. The Realm services table appears.
5. In the DNS config table, click **Add**.
6. **Client realm**—Select a realm from the combination box.

 A screenshot of the 'Add DNS-ALG' dialog box. It has a title bar with a close button. Inside, there is a label '*Client realm' followed by a dropdown menu showing 'access1' and the text '(Unique required)'. At the bottom, there are 'Apply' and 'Cancel' buttons.

7. Click **Apply**.
8. Select the DNS config from the table and click **Edit**.
9. **Description**—Enter the name for the DNS ALG text field. The client realm of this DNS ALG is inherited from its parent and is grayed out.

Description

realm2

10. **DNS service interface IP**—Click **Add**. The Add DNS Service interface IP address dialog box appears.
11. **Client IP address**—Enter the IP address of a client system that can use this DNS ALG.

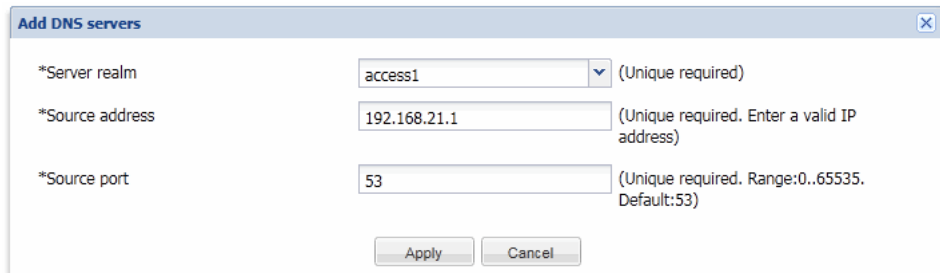
 A screenshot of the 'Add DNS Service interface IP address' dialog box. It has a title bar with a close button. Inside, there is a label '*Client IP address' followed by a text input field containing '111.222.33.34' and the text '(Unique required. Enter a valid IP address)'. At the bottom, there are 'Apply' and 'Cancel' buttons.

12. Click **Apply** to continue. Your entry appears in the DNS service interface window.
13. Select the DNS service interface and click **Edit**.

14. **Transaction Timeout**—Enter the time in seconds to keep DNS mapping information.

Add Domains Served

15. In the DNS servers table, click **Add**. The Add DNS servers dialog box appears.
16. **Server realm**—Choose the server realm from the combination box.
17. **Source address**—Enter the Source address.
18. **Source port**—Enter the Source port.

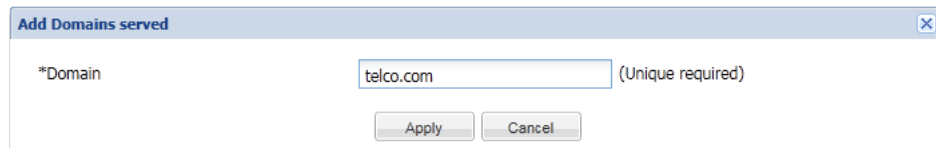


The 'Add DNS servers' dialog box contains three input fields:

- *Server realm**: A dropdown menu with 'access1' selected. A note '(Unique required)' is to the right.
- *Source address**: A text box containing '192.168.21.1'. A note '(Unique required. Enter a valid IP address)' is to the right.
- *Source port**: A text box containing '53'. A note '(Unique required. Range:0..65535. Default:53)' is to the right.

 At the bottom are 'Apply' and 'Cancel' buttons.

19. Click **Apply** to add the entry to the DNS servers table.
20. Select the server from the DNS servers table and click **Edit**.
21. **Domains served**—Click **Add**. The Add DNS Domain served dialog box appears.
22. **Domain**—Enter a domain that this DNS ALG serves.



The 'Add Domains served' dialog box contains one input field:

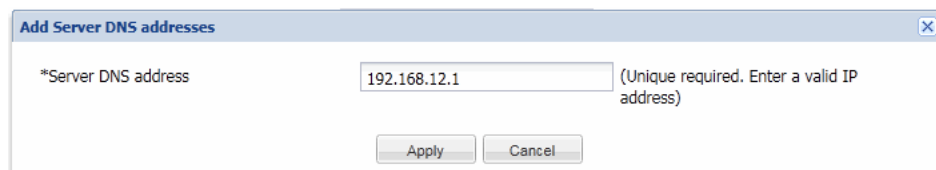
- *Domain**: A text box containing 'telco.com'. A note '(Unique required)' is to the right.

 At the bottom are 'Apply' and 'Cancel' buttons.

23. Click **Apply** to continue. Your entry appears in the Domains served window.

Add DNS Server Addresses

24. **Server DNS address**—Click **Add**. The Add Server DNS Domain served dialog box appears.
25. **Server DNS Address**—Enter a DNS server's IP address.



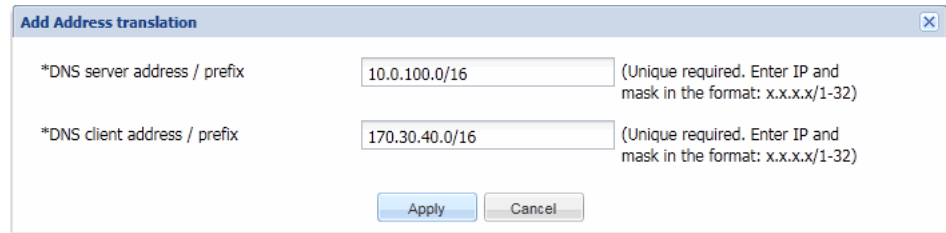
The 'Add Server DNS addresses' dialog box contains one input field:

- *Server DNS address**: A text box containing '192.168.12.1'. A note '(Unique required. Enter a valid IP address)' is to the right.

 At the bottom are 'Apply' and 'Cancel' buttons.

26. Click **Apply** to continue. Your entry appears in the Server DNS address table.
27. Click **Add** in the Address translations section. The Add DNS Address translation dialog box appears.
28. **DNS server address / prefix**—Enter the address/prefix to be returned by the DNS server.

29. **DNS client address / prefix**—Enter the address/prefix to be returned to the client.



The image shows a dialog box titled "Add Address translation" with a close button (X) in the top right corner. It contains two input fields, each with a label and a text box. The first input field is labeled "*DNS server address / prefix" and contains the text "10.0.100.0/16". To the right of this field is a note: "(Unique required. Enter IP and mask in the format: x.x.x.x/1-32)". The second input field is labeled "*DNS client address / prefix" and contains the text "170.30.40.0/16". To the right of this field is the same note: "(Unique required. Enter IP and mask in the format: x.x.x.x/1-32)". At the bottom of the dialog box are two buttons: "Apply" and "Cancel".

*DNS server address / prefix	10.0.100.0/16	(Unique required. Enter IP and mask in the format: x.x.x.x/1-32)
*DNS client address / prefix	170.30.40.0/16	(Unique required. Enter IP and mask in the format: x.x.x.x/1-32)

Apply Cancel

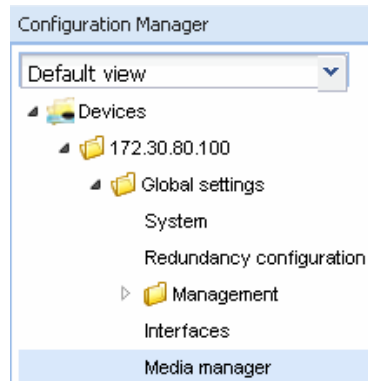
30. Click **Apply** to continue. Entries appear in the Address translations window.
31. Click **Apply** to complete configuring the DNS ALG.

DNS Transaction Timeout

To provide resiliency during DNS server failover, you can now enable a transaction timeout for DNS servers. If you have endpoints that are only capable of being configured with a single DNS server, this can allow DNS queries to be sent to the next configured server—even when contacting the Net-Net SBC's DNS ALG on a single IP address. So when the first server in the list times out, the request is sent to the next server in the list.

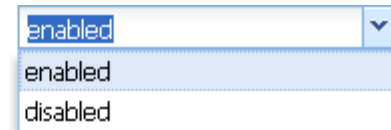
Enabling DNS Transaction Timeout

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. **DNS ALG server failover**—Select enabled from the combination box to enable DNS queries to be sent to the next configured server—even when contacting the Net-Net SBC's DNS ALG on a single IP address. When the first server in the list times out, the request is sent to the next server in the list.

DNS ALG server failover



5. Click **Apply**.

H.248 ALG

You can use the Net-Net SBC to perform the functions of a virtual call agent for H.248 gateways on the access network. On the core side, the H.248 ALG can act as a virtual gateway (one instance for each gateway connecting on the access side) and connect to real call agents. By aggregating signaling and media for many endpoints, it can appear as media gateway controller.

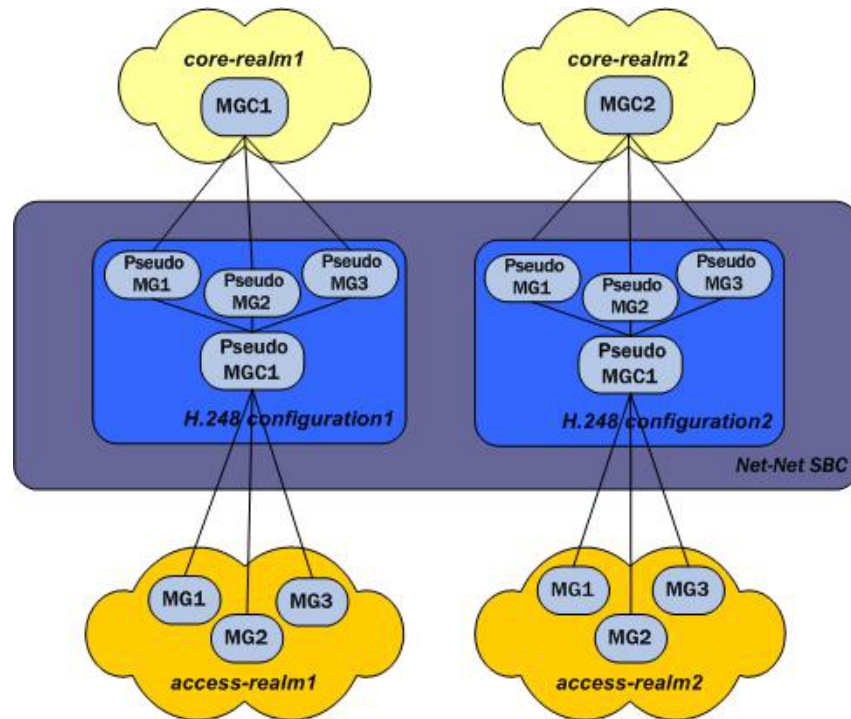
You can configure multiple instances of the virtual call agent and virtual gateway, each with its unique virtual gateway addresses. This way, you can hide topology by keeping external gateways from knowing real call agent addresses.

The H.248 ALG also provides dynamic filters that prevent floods and attacks that use the H.248 protocol. You can define a signaling threshold in bytes per second and then apply the threshold to each gateway individually. If signalling traffic exceeds the threshold, the Net-Net SBC automatically demotes the source IP address.

Note: The Net-Net SBC provides DoS protection on the access side.

Sample Application

In the following diagram, all media gateways sends requests a single IP address and port, which is a virtual (or pseudo) media gateway controller (MGC) on the Net-Net SBC. The Net-Net SBC dynamically created an IP mapping at the core network for each registered gateway. To the core side, the Net-Net SBC appears to be many media gateways (MGs) and connect to the real corresponding softswitch.



Gateway Masquerading

The H.248 supports gateway masquerading, where the Net-Net SBC creates an IP mapping at the core network for each registered gateway. To enable gateway masquerading for the H.248 ALG, you need set an ip-address parameter in the h248-mg-config to a value that has an IP address and a mask.

Consider the following configuration:

```

h248-mgc-config
    name                virtualMgcOne
    ip-address           192.168.32.251
    primary-mg           realMgcOne

h248-mg-config
    mgc-name             realMgcOne
    mgc-ip-address       192.168.232.211
    ip-address           192.168.232.0/25

```

The Net-Net SBC copies all of the host bits—32, minus the netmask from the **IP address** parameter of the H. 248 service/protocol—from the incoming course to the outgoing source. So if the gateway's address arrived at the Net-Net SBC with an incoming value of 192.168.32.10 on the access side, then the outgoing source on the core side would be 192.162.232.10.

Handoff Support

Using an example is the best way to illustrate the Net-Net SBC's handoff support. The example this section discusses uses the following configuration:

```

h248-mgc-config
    name                virtualMgcOne
    ip-address           192.168.32.251
    primary-mg           realMgcOne

h248-mgc-config
    name                virtualMgcTwo
    ip-address           192.168.32.252
    primary-mg           realMgcTwo

h248-mg-config
    mgc-name             realMgcOne
    mgc-ip-address       192.168.232.211
    ip-address           192.168.232.0/25

h248-mg-config
    mgc-name             realMgcTwo
    mgc-ip-address       192.168.232.211
    ip-address           192.168.232.0/25

```

In this example, the gateway on the access side connects to virtualMgcOne. When it receives a registration from the gateway that virtualMgcOne represents, the Net-Net SBC forwards it to the real MGC. It does so using its realMgcOne configuration, which is identified in virtualMgcOne's **Primary MG** setting. From this point, the Net-Net SBC forwards all messages arriving at virtualMgcOne to realMgcOne—and the reverse is also the case.

It is possible, however, that realMgcOne would want to hand off the gateway to realMgcTwo. To do so, it would send a service change message (servicechangeMgcId), containing the IP address and port for realMgcTwo. When it receives the service change message, the Net-Net SBC recognizes that the gateway has been handed off and then selects an H.248 MGC configuration that points to realMgcTwo. As you can see from the sample configuration, it would select virtualMgcTwo. Then the Net-Net SBC forwards the message to the gateway by

modifying the **servicechangeMgcId** so that it contains the IP address for virtualMgcTwo. Now the gateway would register with virtualMgcTwo, and the Net-Net SBC would forward this registration to realMgcTwo.

Licensing

You need to obtain and enable an H.248 license to use the H.248 ALG.

Configuring H.248

This section contains information about how to configure:

- Global H.248 ALG functionality
- H.248 MG
- H.248 MGC

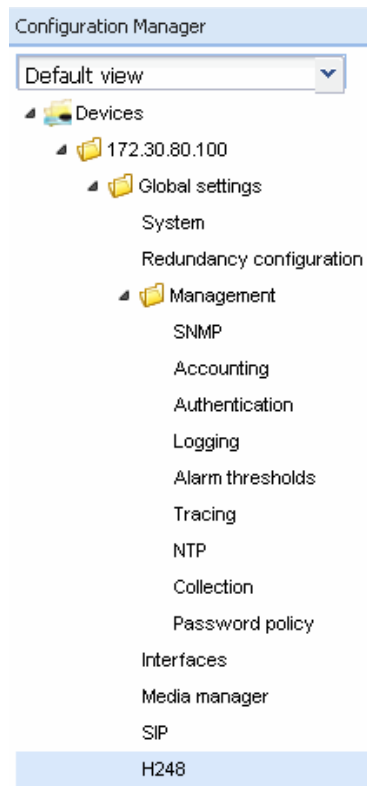
When you enable the H.248 on your Net-Net SBC, keep in mind the following general steps:

6. For the access side, configure an H.248 MGC configuration (**H248MGC-ALG**) to act as the MGC to the gateways on the access side.
7. For the core side, configure an H.248 MG configuration (**H248MG-ALG**) to act as the MG on the core side.
8. Create a bridge between the MG and MGC configurations using the **Primary MG** parameter (in the H.248 configuration); this parameter's value is the **mgc-name** of an **h248-mgc-config**.

Enabling Global H.248 Functionality

To enable global H.248 functionality:

1. Expand the Configuration Manager slider > Global settings.



2. Click H.248.
3. **State**—Retain the default value **enabled** if you want to use the Net-Net SBC's support for H.248. Select **disabled** from the combination box to disable all H.248 functionality.

*State enabled ▼

4. **ALG operation mode**—Select NORMAL in the combination box. This is the only value available.

ALG operation mode normal ▼
normal

5. **Log level**—Select the process log level for monitoring all H.248 activity on the Net-Net SBC in the combination box. The default is INFO.

Note: The system will accept DEBUG, INFO, TRACE, and NOTICE—but they should only be used in consultation with Acme Packet customer support. Any log level you set here overrides the log level you set in the system configuration's process log level parameter.

Log level INFO ▼
MINOR
NOTICE
MAJOR
EMERGENCY
WARNING
DETAIL
CRITICAL
INFO
DEBUG
TRACE

6. **Port to listen for redundancy SYNC messages**—Enter the port on which H.248 checkpointing messages are sent and received. The default value is 1989. A value of 0 disables the H.248 checkpointing. The minimum value is 1025, and the maximum is 65535.
7. **Maximum number of sync transactions to keep on active**—Enter the maximum size of the transaction list, or how many H.248 transactions you want to store in memory at one time. Oldest transactions will be discarded first in the event that the limit is reached. The default value is 10000. The valid range is:
 - Minimum: 0
 - Maximum: 999999999
8. **Timeout for transaction from standby to active**—Enter the number of milliseconds before the Net-Net SBC will try to synchronize its signaling state checkpointing. If the active Net-Net SBC is still adequately healthy, this timer will simply reset itself. If for any reason the active Net-Net SBC has become the standby, it will start to checkpoint with the newly active system when this timer expires.

Acme Packet recommends that you leave this parameter set to its default, 5000. The valid range is:

- Minimum: 0
 - Maximum: 999999999
9. **Timeout for subsequent requests**—Enter the number of milliseconds representing how frequently the standby Net-Net SBC checkpointing with the active Net-Net SBC to obtain the latest H.248 information. The first interval occurs after initial synchronizations of the systems.

Acme Packet recommends that you leave this parameter set to its default, **1000**. The valid range is:

- Minimum: 0
- Maximum: 999999999

Redundancy

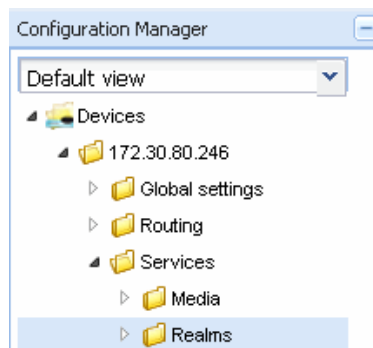
Port to listen for redundancy SYNC messages	1989
Maximum number of sync transactions to keep on active	10000
Timeout for transition from standby to active	5000
Timeout for subsequent requests	1000

10. Click **Apply**.

Configuring the H.248 MGC

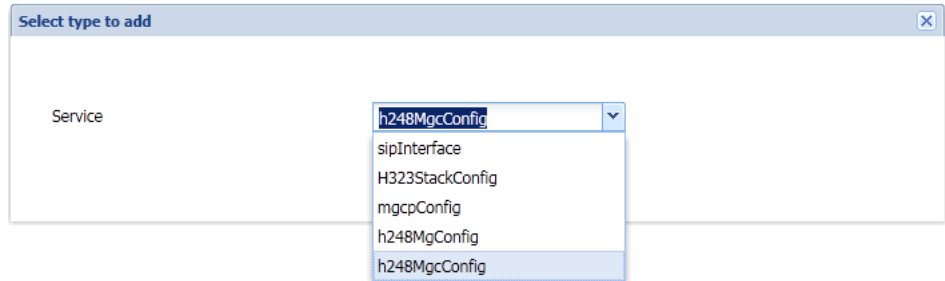
To configure an H.248 MGC:

1. Expand the Configuration Manager slider > Services.



2. Click **Realms**. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. Click the **Services and Protocols** tab. The Realm services table appears.
5. Click **Add**. The Add Realm service dialog box appears.

6. **Service**—Select **h248MgcConfig** in the combination box. The H.248 MGC acts as the MG on the core side.



7. Click **Apply**. The Add Services and Protocols dialog box appears.
8. **Name**—Enter the hostname of the MGC. You use this value to in the **Primary MG** parameter when you want to bridge between an MGC configuration and an MG configuration. This parameter has no default, but it is required.
9. **IP address**—Enter the IP address of the real MGC on the core side. This parameter has no default, but it is required.
10. Select the MGC service from the table and click **Edit**.
11. **Primary MG**—Click the name value in the combination box that you want to associate with this MGC configuration. This parameter's value (which is a hostname) creates the bridge between the MGC and the MG configurations. This parameter is required, but it has no default value.
12. Click **OK**. The Edit H248 ALG service dialog box appears.
13. **State**—Retain the default value **enabled** if you want to use this H.248 MGC configuration. Select **enabled** from the combination box to **disable** H.248 MGC.
14. **IP address**—Enter the IP address for the Net-Net SBC to use when acting as an MGC on the access side. This parameter is required, but it has no default value.
15. **Port**—Enter the port number you want to be associated with the IP address you entered for the **IP address** parameter. The default value is **2944**. The minimum value for this parameter is **1025**, and the maximum is **65535**.
16. **Primary MG**—Select the name value in the combination box that you want to associate with this MGC configuration. This parameter's value (which is a hostname) creates the bridge between the MGC and the MG configurations. This parameter is required, but it has no default value.
17. **Transport method**—Select the transport method you want to use in the combination box. The default is **UDP**, and only UDP is supported at this time.
18. **Encoding**—Select the encoding method you want to use for this MGC in the combination box. You can use either **text** (default) or **binary**.
19. **NAT traversal**—Retain the default value **disabled**, or select **enabled** from the combination box to **enable** NAT traversal to signify that gateway being referenced in this configuration is behind NATs.

20. **Audit interval**—Enter the time in seconds you want to use as the interval between AuditValue messages. The default for this parameter is 0..

[realmConfig->h248MgcConfig](#)

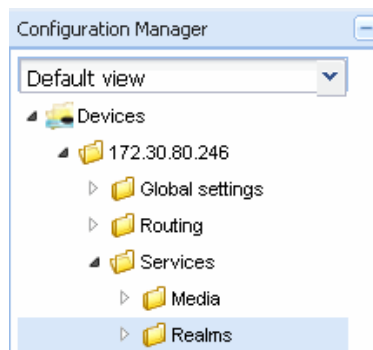
State	enabled
*Name	MGC hostname
*Realm ID	access1
IP address	172.55.55.55
Port	2944
Primary MG	
Transport method	UDP
Encoding	text
Nat traversal	disabled
Audit interval	0

21. Click **Apply**.

Configuring the H.248 MG

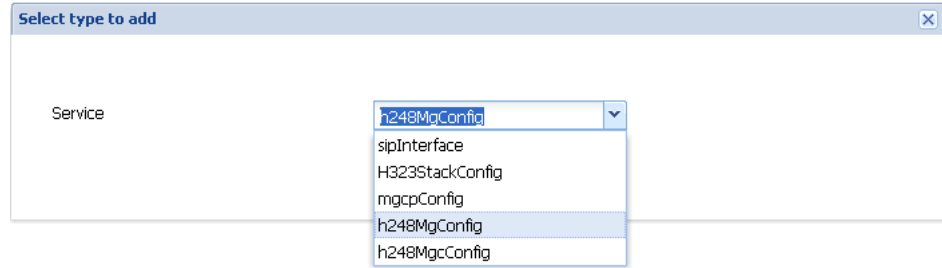
To configure an H.248 MG:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Click **Add**. The Add Realm service dialog box appears.

6. **Service**—Select **h248MgConfig** in the combination box. The H.248 MG acts as the MG on the access side.



7. Click **Apply**. The Add Services and Protocols dialog box appears.
8. **Name**—Enter the hostname of the MG. You use this value in the **Primary MG** parameter when you want to bridge between an MG configuration and an MG configuration. This parameter has no default, but it is required.
9. **IP address**—Enter the IP address of the real MG on the access side. This parameter has no default, but it is required.
10. Select the MG service from the table and click **Edit**.
11. **Encoding**—Select the encoding method you want to use for this MGC in the combination box. You can use either **text** (default) or **binary**.
12. **MGC port**—Enter the port number you want to be associated with the IP address you entered for the **IP address** parameter. The default value is **2944**. The minimum value for this parameter is **1025**, and the maximum is **65535**.
13. Click **Apply**.

DNS Transaction Timeout

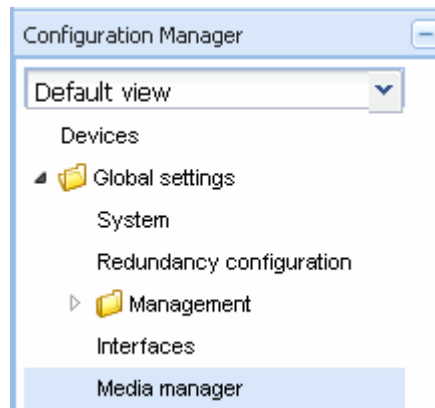
To provide resiliency during DNS server failover, you can now enable a transaction timeout for DNS servers. If you have endpoints that are only capable of being configured with a single DNS server, this can allow DNS queries to be sent to the next configured server—even when contacting the Net-Net SBC's DNS ALG on a single IP address. So when the first server in the list times out, the request is sent to the next server in the list.

The Net-Net SBC uses the transaction timeout value set in the **DNS server attributes** configuration (part of the DNS configuration).

Enabling DNS Transaction Timeout

To enable the DNS transaction timeout:

1. Expand the Configuration Manager slider -> Global settings.



2. Click Media Manager.
3. Scroll to **Enable DNS ALG server failover**.
4. **DNS ALG server failover**—Click **enabled** in the drop down list to allow DNS queries to be sent to the next configured server—even when contacting the Net-Net SBC's DNS ALG on a single IP address. So when the first server in the list times out, the request is sent to the next server in the list.

DNS ALG server failover enabled

5. Click **Apply**.

Recursive ENUM Queries

If the Net-Net SBC receives an A-record in response to an ENUM query, it will reperform that ENUM query to the server received in the A-record.

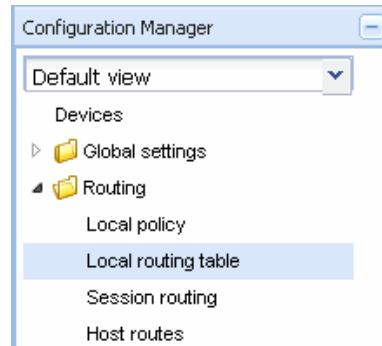
If the Net-Net SBC receives an NS record in response to an ENUM query, it will resend the original ENUM query to the DNS server defined in the realm of the FQDN in the NS record. It will use the response to perform a subsequent ENUM query.

This behavior is configured by setting the **recursive query** parameter in the enum config to **enabled**.

Configuring Recursive Query

To configure the Net-Net SBC to retarget or route request messages when performing an ENUM lookup:

1. Expand the Configuration Manager slider -> Routing.
2. Click Local routing table.



3. Click the local routing config you want to configure in the table and click **Edit**.
4. **Recursive query**—Click **enabled** in the drop down list for the Net-Net SBC to query a DNS server for a hostname returned in an ENUM result.
5. Click **Apply**.

Unique DNS Server per VLAN with OLIP

There is no configuration required for this feature. It is enabled automatically.

DNS ALG Message Throttling

You can limit DNS ALG-bound throughput by using the `dns-alg-constraints` configuration element. Message throttling is performed on request messages, and since DNS-ALG is transaction stateful, the responses are automatically throttled. Once a `dns-alg-constraints` element is configured, it can be referenced by a DNS configuration object. This allows users to create constraint profiles and apply them to multiple DNS configuration objects.

When any of the constraints are exceeded, the DNS server's status changes to Constraints Exceeded and requests are rejected with a 503 error message. The server, via the ALG, remains in that state until the time-to-resume parameter period ends. When this period ends, the Net-Net DD re-examines the server's traffic with respect to the current burst or sustain window and the state is determined again, along with the decision to readmit traffic.

Bursty Traffic Throttling

You can create separate inbound and outbound maximum burst rates. First configure a **Burst window** in seconds. Then configure the **Maximum inbound burst rate** and **max outbound burst rate** parameters for the traffic you wish to constrain. In addition you can set an overall maximum burst rate with the **max burst rate** parameter.

Regardless of the inbound or outbound burst rates' headroom before exceeding the message constraints, if the total in/out traffic exceeds the **max burst rate** (when

configured), the target DNS server is set to Constraints Exceeded status and taken out of service.

Sustained Traffic Throttling

You can create separate inbound and outbound maximum sustained traffic rates. First configure a **Sustained window** in seconds. Then configure the **Maximum inbound sustain rate** and **Maximum outbound sustain rate** parameters. In addition you can set an overall sustain rate with the **Maximum sustain rate** parameter.

Regardless of the inbound or outbound sustained traffic rates' headroom before exceeding the message constraints, if the total in/out traffic exceeds the **Maximum sustained rate** (when configured), the target DNS server is set to Constraints Exceeded status and taken out of service.

Maximum Latency

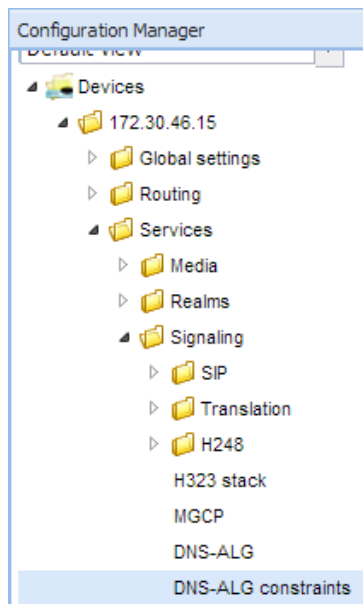
The Maximum latency parameter sets a threshold under which requests must be responded to. If this value is exceeded, the DNS server is put into the Constraints Exceeded State.

Configuration Instructions

Global Constraints

To configure the DNS ALG Constraints object:

1. Expand the Configuration Manager slider > Services > Signaling.



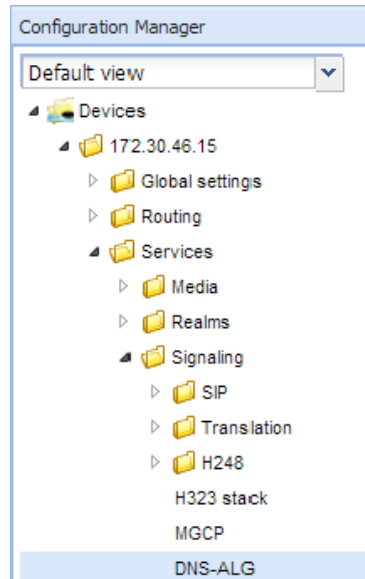
2. Click DNS-ALG constraints.
3. Click **Add**.
4. **Name**—Enter a name for this DNS-ALG constraints configuration element. You will reference this value from the constraint-name parameter in the DNS-ALG configuration element.
5. Click **Apply**.

6. Select the DNS-ALG constraints configuration from the table and click **Edit**.
7. **State**—Set this parameter to enabled to enable this configuration element.
8. **Burst window**—Enter the number of seconds during which to count messages toward a maximum burst rate.
9. **Maximum burst rate**—Enter the maximum number of messages that can pass through the system in the burst rate window before setting the element to Constraints Exceeded.
10. **Maximum inbound burst rate**—Enter the maximum number of inbound messages received by the referencing element within the burst rate window before setting the element to Constraints Exceeded.
11. **Maximum inbound sustain rate**—Enter the maximum number of inbound messages received by the referencing element within the sustained rate window before setting the element to Constraints Exceeded.
12. **Maximum latency**—Enter the time the Net-Net SBC waits to receive a response to a request. After this value is exceeded, the DNS server is placed in Constraints Exceeded state.
13. **Maximum outbound burst rate**—Enter the maximum number of outbound messages forwarded from the referencing element within the burst rate window before setting the element to Constraints Exceeded.
14. **Maximum outbound sustain rate**—Enter the maximum number of outbound messages forwarded from the referencing element within the sustained rate window before setting the element to Constraints Exceeded.
15. **Maximum sustain rate**—Enter the maximum number of messages that can pass through the system in the sustained rate window before setting the element to Constraints Exceeded.
16. **Sustained window**—Enter the number of seconds during which to count messages toward a maximum sustained rate.
17. **Time to resume**—Enter the number of seconds that the referencing element stays in Constraints Exceeded state and rejects messages before it returns to service.
18. Click **Apply**.

Applying Traffic Constraints

To apply DNS ALG constraints to an existing DNS config object:

1. Expand the Configuration Manager slider > Services > Signaling.



2. Click DNS-ALG.
3. **Constraint name**—Select the name of an existing DNS-ALG constraints configuration element.
4. Click **Apply**.

10 Session Routing and Load Balancing

Introduction

This chapter explains how to configure session routing and load balancing for SIP and H.323 services. It contains information about configuring session agents and session agent groups, as well as local policies that can be used for routing SIP or H.323 signals.

Routing Overview

This section provides an overview of routing SIP and H.323 sessions when using the Net-Net SBC. The Net-Net SBC chooses the next hop through the network for each SIP and H.323 session based on information received from routing policies and constraints. Routing policies can be as simple as routing all traffic to a proxy or routing all traffic from one network to another. Routing policies can also be more detailed, using constraints to manage the volume and rate of traffic that can be routed to a specific network. For example, you can manage volume and rate of traffic to enable the Net-Net SBC to load balance and route around softswitch failures.

When a call request arrives at the Net-Net SBC, a decision making process then occurs to determine whether the message is coming from a session agent. If so, the Net-Net SBC checks whether that session agent is authorized to make the call. Local policy is then checked to determine where to send the message on to.

Session Agents, Session Groups, and Local Policy

When you configure session routing for SIP and H.323, you can use session agents, session agent groups and local policies to define routing. (Using session agents and session agent groups is not required.)

- session agent: defines a signaling endpoint. It is a next hop signaling entity that can be configured to apply traffic shaping attributes.
- session agent group (SAG): can contain individual session agents and other session agent groups. Members of a SAG are logically equivalent (although they might vary in their individual constraints) and can be used interchangeably.

You apply an allocation strategy to the SAG to allocate traffic across the group members. Session agent groups also assist in load balancing among session agents.

- local policy: indicates where session request messages, such as SIP INVITES, are routed and/or forwarded. You use a local policy to set a preference for selecting one route over another.

Another element of routing is the realm. Realms are used when a Net-Net SBC communicates with multiple network elements over a shared intermediate connection. Defining realms allows sessions to go through a connection point between the two networks. See *Configuring Realms* for additional details.

When you configure a realm, you give it an identifier, which stores the name of the realm associated with the Net-Net system. The realm identifier value is also needed when you configure session agents and local policies. You can associate a realm with a session agent to identify the realm for sessions coming from or going to the session agent. You also need the realm identifier when you configure local policy to identify the egress realm (realm of the next hop).

About Session Agents

This section describes session agents. A session agent defines a signaling endpoint. It is a next hop signaling entity that can be configured to apply traffic shaping attributes. Service elements such as gateways, softswitches, and gatekeepers are defined automatically within the Net-Net SBC as session agents. For each session agent, concurrent session capacity and rate attributes can be defined. You can group session agents together into session agent groups and apply allocation strategies to achieve traffic load balancing.

You can assign a media profile to a session agent and indicate whether the transport protocol is SIP or H.323. If the protocol is H.323, you need to indicate whether the session agent is a gateway or a gatekeeper.

You can configure a set of attributes and constraints for each session agent to support the following:

- session access control: Net-Net SBC only accepts requests from configured session agents
- session admission control (concurrent sessions): Net-Net SBC limits the number of concurrent inbound and outbound sessions for any known service element.
- session agent load balancing: session agents are loaded based on their capacity and the allocation strategy specified in the session agent group.
- session (call) gapping: Net-Net SBC polices the rate of session attempts to send to and receive from a specific session agent.

SIP Session Agents

SIP session agents can include the following:

- softswitches
- SIP proxies
- application servers
- SIP gateways
- SIP endpoints

In addition to functioning as a single logical next hop for a signaling message (for example, where a SIP INVITE is forwarded), session agents can provide information about next or previous hops for packets in a SIP agent, including providing a list of equivalent next hops.

You can use the session agent to describe one or more SIP next or previous hops. Through the configured carriers list, you can identify the preferred carriers to use for traffic coming from the session agent. This set of carriers will be matched against the local policy for requests coming from the session agent. You can also set constraints for specific hops.

H.323 Session Agents

H.323 session agents can include the following:

- Gatekeepers
- Gateways
- MCUs

Managing Session Agent Traffic

The Net-Net SBC monitors availability, session load, and session rate for each session agent in real time. The session agent's state is determined by its performance relative to the constraints applied to it and its availability.

The following table lists the conditions that cause the Net-Net SBC to suspend the routing of traffic to a session agent, along with the criteria for restoring the route.

Constraint Condition	SIP Criteria	H.323 Criteria	Action	Criteria for Resuming
Maximum sessions exceeded	Maximum concurrent SIP sessions exceeded.	Maximum concurrent H.323 sessions exceeded. If the session agent is a gatekeeper and gatekeeper routed mode is not used, this constraint is an aggregate of all the destination gateways. Only maximum outbound sessions are measured.	Session agent is declared in constraint violation state.	Concurrent sessions drop below the maximum sessions value.
Maximum outbound sessions exceeded	Maximum concurrent outbound SIP sessions exceeded.	Maximum concurrent outbound H.323 sessions exceeded. If the session agent is a gatekeeper and gatekeeper routed mode is not used, this constraint is an aggregate of all the destination gateways. Only maximum outbound sessions are measured.	Session agent is declared in constraint violation state.	Concurrent sessions drop below the maximum outbound sessions value.
Maximum burst rate exceeded	Maximum burst rate exceeded in current window.	Maximum burst rate exceeded in current window. If the session agent is a gatekeeper and gatekeeper routed mode is not used, this constraint is an aggregate of all the destination gateways. Only maximum outbound sessions are measured.	Session agent is declared in constraint violation state.	Burst rate in subsequent window drops below maximum burst rate.

Constraint Condition	SIP Criteria	H.323 Criteria	Action	Criteria for Resuming
Maximum sustained rate exceeded	Maximum sustained rate exceeded in current window.	Maximum burst rate exceeded in current window. If the session agent is a gatekeeper and gatekeeper routed mode is not used, this constraint is an aggregate of all the destination gateways. Only maximum outbound sessions are measured.	Session agent is declared in constraint violation state.	Sustained rate in subsequent window drops below the maximum sustained rate.
Session agent unavailable or unresponsive	SIP transaction expire timer expires for any out-of-dialogue request. For example, INVITE, REGISTER, or ping.	- Response timer expires. The default is T301=4 seconds. - Connect timer expires. The default is T303=32 seconds. If the session agent is a peer gatekeeper, the LRQ response time is used to determine availability. The RAS response timer is 4 seconds.	Session agent is declared in constraint violation state or out-of-service. The time to resume timer starts.	Time to resume timer expires and the Net-Net SBC declares the session agent in-service. or Session agent responds to subsequent pings (SIP only).

About Session Agent Groups

Session agent groups can contain individual session agents and other session agent groups. Members of a session agent group are logically equivalent (although they might vary in their individual constraints) and can be used interchangeably. You can apply allocation strategies to session agent groups.

Examples of session agent groups include the following:

- application server cluster
- media gateway cluster
- softswitch redundant pair
- SIP proxy redundant pair
- gatekeeper redundant pair

Session agent group members do not need to reside in the same domain, network, or realm. The Net-Net SBC can allocate traffic among member session agents regardless of their location. It uses the allocation strategies configured for a SAG to allocate traffic across the group members.

Allocation strategies include the following:

Allocation Strategy	Description
Hunt	Net-Net SBC selects the session agents in the order in which they are configured in the SAG. If the first agent is available, and has not exceeded any defined constraints, all traffic is sent to the first agent. If the first agent is unavailable, or is in violation of constraints, all traffic is sent to the second agent. And so on for all session agents in the SAG. When the first agent returns to service, the traffic is routed back to it.
Round robin	Net-Net SBC selects each session agent in the order in which it is configured, routing a session to each session agent in turn.
Least busy	Net-Net SBC selects the session agent with the least number of active sessions, relative to the maximum outbound sessions or maximum sessions constraints (lowest percent busy) of the session agent.
Proportional distribution	Session agents are loaded proportionately based upon the respective maximum session constraint value configured for each session agent.
Lowest sustained rate	Net-Net SBC routes traffic to the session agent with the lowest sustained session rate, based on observed sustained session rate.

You apply allocation strategies to select which of the session agents that belong to the group should be used. For example, if you apply the Hunt strategy session agents are selected in the order in which they are listed.

About Local Policy

This section explains the role of local policy. Local policy lets you indicate where session requests, such as SIP INVITES, should be routed and/or forwarded. You use a local policy to set a preference for selecting one route over another. The local policy contains the following information that affects the routing of the SIP and H.323 signaling messages:

- Information in the From header
Information in the message's From header is matched against the entries in the local policy's from address parameter to determine if the local policy applies.
- List of configured realms
This list identifies from what realm traffic is coming and is used for routing by ingress realm. The source realms identified in the list must correspond to the valid realm IDs you have already configured
- Local policy attributes
The attributes serve as an expression of preference, a means of selecting one route over another. They contain information such as the next signaling address to use (next hop) or whether you want to select the next hop by codec, the realm of the next hop, and the application protocol to use when sending a message to the next hop. You can also use the attributes to filter specific types of traffic.

Routing Calls by Matching Digits

Local policy routing of a call can be based on matching a sequence of digits against what is defined in the local policy. This sequence refers to the first digits in the (phone) number, matching left to right.

The following examples show how the Net-Net system matches an area code or number code against configured local policies.

- If the number or area code being matched is 1234567 (where 123 is an area code), and the from address value in one local policy is 123, and the from address value in another local policy is 12, the Net-Net system forwards the call to the server that is defined as the next hop in the local policy with 123 as the from address value.
- If the number or area code being matched is 21234, and the from address value in one local policy is 123, and the from address value in another local policy is 12, the Net-Net system will not find a match to either local policy because the first character of the number or area code must match the first character in a from address or to address field.

The following examples show how the Net-Net system matches an area or number code against different local policies: the first one has a From address value of 12 and the second has a From address value of 123. The Net-Net system chooses the route of the local policy that is configured with the most digits matching the area or number code in its From address and To address fields.

- When the two different local policies route to two different servers, and the area or number code being matched is 123, the Net-Net system selects the second local policy based on the From address value of 123.
- When the two different local policies route to two different servers, and the area or number code being matched is 124, the Net-Net system selects the first local policy based on the From address value of 12.

SIP and H.323 Interworking

You need to configure local policies, including the requisite local policy attributes, to use the H.323<—>SIP interworking (IWF). Flow progression in H.323<—>SIP traffic depends heavily on the local policies configured for the Net-Net SBC, which determine what protocol is used on the egress side of a session.

You set the application protocol (an local policy attribute option) to instruct the Net-Net SBC to interwork the protocol of an ingress message into a different protocol (H.323<—>SIP or SIP—>H.323) upon its egress to the next hop.

For example, if the application protocol is set to SIP, an inbound H.323 message will be interworked to SIP as it is sent to the next hop. An inbound SIP message would pass to the next hop unaffected. If the application protocol is set to H.323, an inbound SIP message will be interworked to H.323 before being sent to the next hop.

See *Configuring SIP and H.323 IWF Signaling* for more information.

Route Preference

The Net-Net SBC builds a list of possible routes based on the source realm and the From-address (From-URI) and To-address (Request-URI), which forms a subset from which preference then decides. Any local policy routes currently outside of the configured time/day are not used, if time/day are set. Also, any local policy routes not on the list of carriers (if carriers is set and the requests has a Carrier header) are not used.

The Net-Net SBC applies preference to configured local policies in the following order:

Note: Source realm is used in the local policy lookup process, but it is not used in route preference calculations. The Net-Net SBC applies preference to configured local policies in the following order:

1. Cost (cost option in local policy attributes) is always given preference.
2. Matching media codec (media profiles option in local policy attributes).
3. Longest matching To address (To address list in local policy).
4. Shortest matching To address (To address list in local policy).
5. Longest matching From address (from address list in local policy).
6. Shortest matching From address (from address list in local policy).
7. Narrowest/strictest day of week specification (days of week option in local policy attributes).
8. Narrowest/strictest time of day specification (start time and end time options in local policy attributes).
9. Wildcard matches (use of an asterisk as a wildcard value for the from address and to address lists in local policy).
10. Wildcard card matches are given the least preference. A prefix value of 6 is given a higher preference than a prefix value of * even though both prefix values are, in theory, the same length.

SIP Session Routing

This section describes SIP session routing. When routing SIP call requests, the Net-Net SBC communicates with other SIP entities, such as SIP user devices, other SIP proxies, and so on, to decide what SIP-based network resource each session should visit next. The Net-Net SBC processes SIP call requests and forwards the requests to the destination endpoints to establish, maintain, and terminate real-time multimedia sessions.

Certain items in the messages are matched with the content of the local policy, within constraints set by the previous hop session agent, and the SIP configuration information (for example, carrier preferences) to determine a set of applicable next hop destinations.

The sending session agent is validated as either a configured session agent or a valid entry in a user cache. If the session INVITATION does not match any registering user, the SIP proxy determines the destination for routing the session INVITATION.

Limiting Route Selection Options for SIP

You can configure the local policy to use the single most-preferred route. And you can configure the SIP configuration max routes option to restrict the number of routes which can be selected from a local policy lookup:

- A `max-routes=1` value limits the Net-Net SBC to only trying the first route from the list of available preferred routes.
- A `max-routes=0` value or no `max-routes` value configured in the `options` field allows the Net-Net SBC to use all of the routes available to it.

A Net-Net SBC configured for H.323 architectures will have access to all of the routes it looks up by default.

About Loose Routing

According to RFC 3261, a proxy is loose routing if it follows the procedures defined in the specification for processing of the Route header field. These procedures

separate the destination of the request (present in the Request-URI) from the set of proxies that need to be visited along the way (present in the Route header field).

When the SIP NAT's route home proxy field is set to enabled, the Net-Net SBC looks for a session agent that matches the home proxy address and checks the loose routing field value. If the loose routing is:

- enabled: a Route header is included in the outgoing request in accordance with RFC 3261.
- disabled: a Route header is not included in the outgoing request; in accordance with the route processing rules described in RFC 2543 (referred to as strict routing). That rule caused proxies to destroy the contents of the Request-URI when a Route header field was present.

Whether the loose routing field is enabled is also checked when a local policy's next hop value matches a session agent. Matching occurs if the hostname or the session agent's IP address field value corresponds to the next hop value. If loose routing is enabled for the matching session agent, the outgoing request retains the original Request-URI and Route header with the next hop address.

About the Ingress Realm

You can create a list of realms in your local policy that is used by the Net-Net SBC to determine how to route traffic. This list determines from which realm traffic is coming and is used for routing by the ingress realm. The source realm values must correspond to a valid identifier entered when the realm was configured.

About the Egress Realm

An egress realm allows SIP signaling to travel out of the Net-Net system through a network other than the home realm. The Net-Net system uses egress realms for signaling purposes (when matching flows). When a packet arrives at the Net-Net system with a destination address that does not match any defined session agents, the Net-Net system uses the address associated with the realm that is, in turn, associated with the SIP configuration's egress realm ID, as the outgoing network. With the use of the egress realm ID, it is possible to define a default route for SIP requests addressed to destinations outside the home realm. If no egress realm is defined, the home realm (default ingress realm) is used as the default egress realm.

About SIP Redirect

SIP redirect involves proxy redirect and tunnel redirect.

Proxy Redirect

You can configure the SIP proxy mode to define how the SIP proxy will forward requests coming from the session agent. This value is used if the session agent's proxy mode has no value (is empty).

Tunnel Redirect

You can use tunnel redirect when requests are routed to a server behind a SIP NAT that sends redirect responses with addresses that should not be modified by the SIP NAT function. For example, a provider might wish to redirect certain calls (like 911) to a gateway that is local to the UA that sent the request. Since the gateway address is local to the realm of the UA, it should not be modified by the SIP NAT of the server's realm. Note that the server must have a session agent configured with the redirect-action field set to the proxy option in order to cause the redirect response to be sent back to the UA.

SIP Method Matching and To Header Use for Local Policies

For SIP, this feature grants you greater flexibility when using local policies and has two aspects: basing local policy routing decisions on one or more SIP methods you configure and enabling the Net-Net SBC to use the TO header in REGISTER messages for routing REGISTER requests.

SIP Methods for Local Policies

This feature allows the Net-Net SBC to include SIP methods in routing decisions. If you want to use this feature, you set a list of one or more SIP methods in the local policy attributes. These are the SIP methods you can enter in the list: INVITE, REGISTER, PRACK, OPTIONS, INFO, SUBSCRIBE, NOTIFY, REFER, UPDATE, MESSAGE, and PUBLISH.

After the Net-Net SBC performs a local policy look-up for SIP, it then searches for local policy attributes that have this methods list configured. If it finds a set of policy attributes that matches a method that matches the traffic it is routing, the Net-Net SBC uses that set of policy attributes. This means that the Net-Net SBC considers first any policy attributes with methods configured before it considers those that do not have methods. In the absence of any policy attributes with methods, the Net-Net SBC uses the remaining ones for matching.

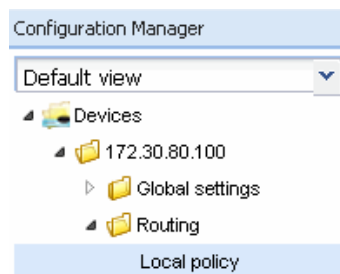
In cases where it finds neither matching policy attributes with methods or matching policy attributes without them, the Net-Net SBC either rejects the calls with a **404 No Routes Found** (if the request calls for a response) or drops the call.

Configuring SIP Methods for Local Policy Matching

You configure local policy matching with SIP methods in the local policy attributes parameter calls **methods**. This parameter is a list that takes either one or multiple values.

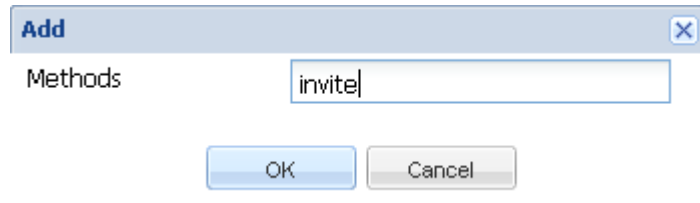
To configure SIP methods for local policy matching:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Click a policy name in the table and click **Edit**.
4. Scroll to the Route policy attributes table, select an attribute and click **Edit**.
5. Scroll to Methods.
6. Click **Add**. The Methods dialog box appears.
7. **Methods**—Enter the SIP method you want to use for matching this set of policy attributes. By default, this parameter is empty—meaning that SIP methods will not be taken into consideration for routing based on this set of policy attributes.
 - INVITE

- REGISTER
- PRACK
- OPTIONS
- INFO
- SUBSCRIBE
- NOTIFY
- REFER
- UPDATE
- MESSAGE
- PUBLISH



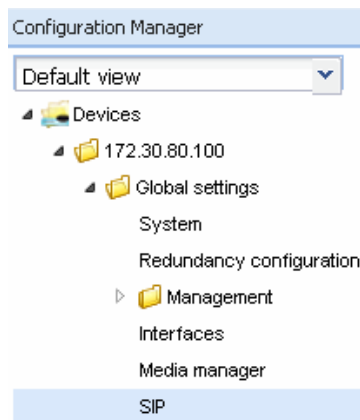
- Click **OK**.
- Repeat steps 6 through 9 if you want to enter more than one method.
- Click **Apply**.

Configuring TO Header Use

For the Net-Net SBC's global SIP configuration, you can enable the use of an ENUM query to return the SIP URI of the Registrar for a SIP REGISTER message. Without this feature enabled, the Net-Net SBC uses the REQUEST URI. This ability can be helpful because REGISTER messages only have the domain in the REQUEST URI, whereas the SIP URI in the To header contains the user's identity.

To configure TO header use for local routing:

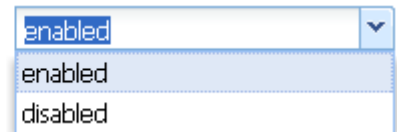
- Expand the Configuration Manager slider > Global settings.



- Click SIP.
- Click **Add** to add the SIP configuration in the content area.
- Scroll to Use To: header for local policy match on register.

5. **Use To: header for local policy match on register**—Select enabled from the combination box to enable this parameter if you want the Net-Net SBC to use an ENUM query to return the SIP URI of the Registrar for a SIP REGISTER message. This parameter defaults to **disabled**.

Use To: header for local policy match



enabled	▼
enabled	
disabled	

6. Click **Apply**.

H.323 Session Routing

This section describes H.323 session routing.

Egress Stack Selection

Egress stack selection includes static stack selection and policy-based stack selection.

Static Stack Selection

In static stack selection, the outgoing stack is determined through the establishment of associated stacks in the h323 stack.

The incoming stack (configured in the h323 stack) uses its associated stack value to determine the associated outgoing stack. The associated stack value corresponds to the name of an h323 stack. This type of selection is referred to as *static* because the incoming stack always uses the stack specified in the associated stack as the outgoing stack; no other stacks are considered.

Policy-Based Stack Selection

The Net-Net SBC performs dynamic, policy-based stack selection when an H.323 call arrives at the Net-Net SBC and a configured associated outgoing stack cannot be found.

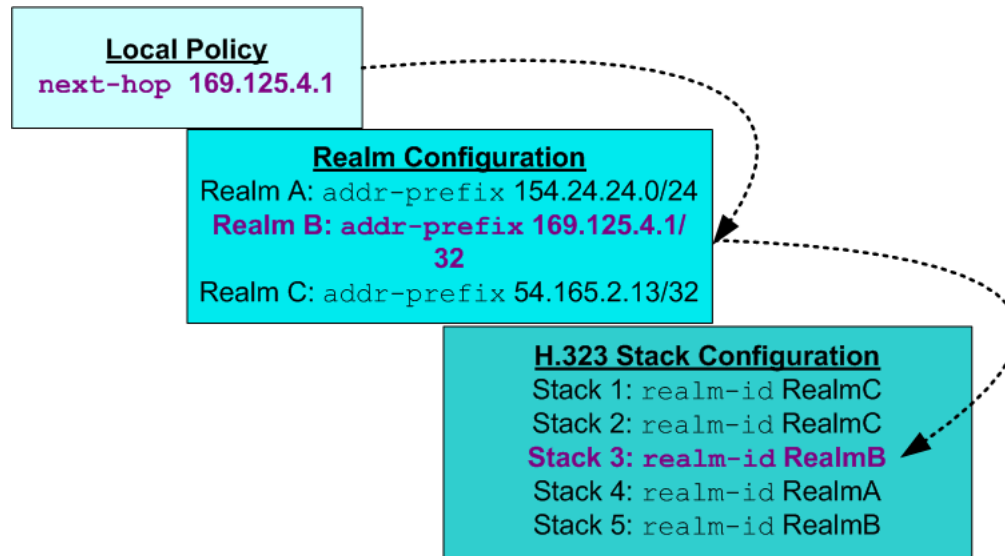
For policy-based stack selection, the Net-Net SBC refers to local policies that contain address information that corresponds to incoming traffic. This information is contained in the local policy's To address and From address fields. For the source, this information is matched with the Q.931 calling party number; if there is no calling party number, the H.323 source address is used. For the destination, this information is matched with the called party number; if there is no called party number, then the H.323 destination address is used.

After a local policy corresponding to the incoming traffic has been found, the Net-Net SBC looks at the next hop value (a local policy attribute) and selects a local policy for the basis of stack selection. If the local policy look-up yields multiple local policies with the same next hop values, but with different `cost` values, the local policy with the lowest `cost` value is selected.

If a realm is not defined in the local policy, the next hop address is then matched against the address prefix values for the realms that are configured for the system. Thus, the Net-Net SBC discovers the realm for this traffic. Using this realm information, the Net-Net SBC performs stack selection. It uses the first configured H.323 stack in the Net-Net SBC's configuration that has a realm ID value matching the identifier field of the realm with the appropriate address prefix.

In the following example, the local policy matching yields a local policy with a next hop value of `169.125.4.1`, which corresponds to RealmB. The outgoing stack

selected is Stack 3 because it is the first stack to have been configured with RealmB as the realm ID.



Registration Caching

The Net-Net SBC can cache and proxy an H.225 RegistrationRequest (RRQ) between an H.323 endpoint and a gatekeeper. Registration caching serves two functions:

- It allows aggregation of RRQs sent to a gatekeeper stack and proxies those requests through the gateway stack. If the external gatekeeper associated with the gatekeeper stack supports additive registration, the requests will be consolidated. Furthermore, if the gatekeeper supports additive registration, the Net-Net SBC will register in an additive manner, meaning that will send additive RRQs.
- It allows the gatekeeper stack to use the registration information to route calls from other realms to endpoints in its realms.

To perform registration caching, the Net-Net SBC must be configured with at least two stacks. One of these stacks will receive registrations (*gatekeeper stack*), and one stack will proxy registrations (*gateway stack*). The Net-Net SBC caches all successful registrations and uses the cache to route calls to the endpoints.

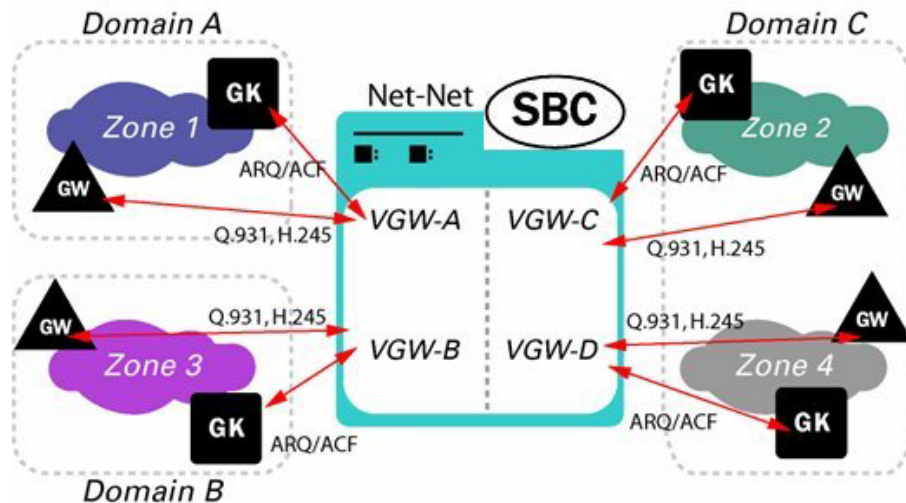
Gatekeeper Provided Routes

Gatekeeper provided routes include back-to-back gateways, back-to-back gatekeeper and gateway, and interworking gatekeeper/gateway.

Back-to-Back Gateway

When the Net-Net SBC is functioning as a back-to-back gateway (B2BGW), it appears as multiple H.323 gateways to multiple networks. Each Net-Net SBC virtual gateway discovers and registers with a gatekeeper in its respective domain. Each gateway relies on its gatekeeper for admission and location services through the ARQ/ACF exchange. H.225 call control and H.245 messages are exchanged directly with the terminating gateway or gatekeeper. Routing policies are used to associate one virtual gateway with another.

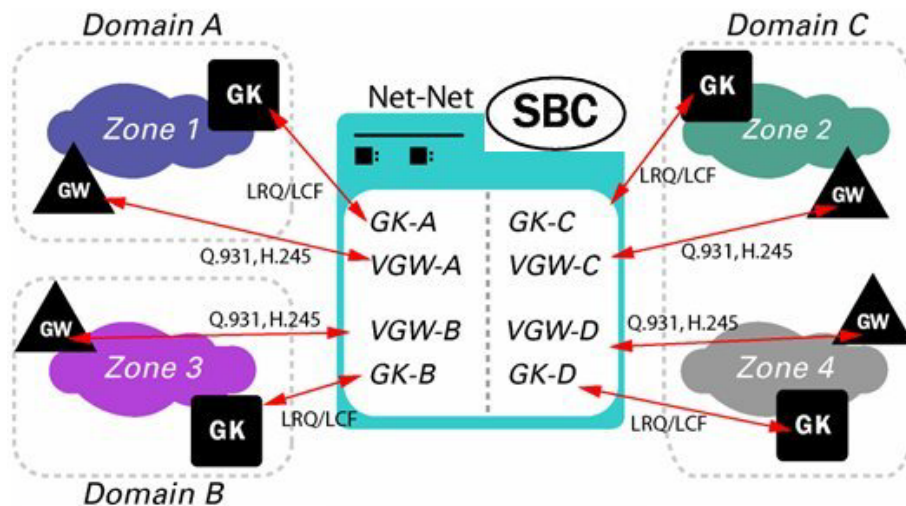
The following diagram illustrates the back-to-back gateway.



Back-to-Back Gatekeeper and Gateway

For peering connections where both networks use inter-domain gatekeeper signaling, the Net-Net SBC is configured as a back-to-back gatekeeper proxy and gateway mode of operation. The Net-Net SBC responds and issues LRQs and LCFs/LRJs acting as a routed gatekeeper. Peered gatekeepers send LRQ to the RAS address of one of the Net-Net SBC's virtual gatekeepers and it responds by providing its call signaling address that performs the gateway functions. Routing policies are used to determine the egress virtual gatekeeper that then exchanges LRG/LCF to determine the call signaling address of the terminating gateway.

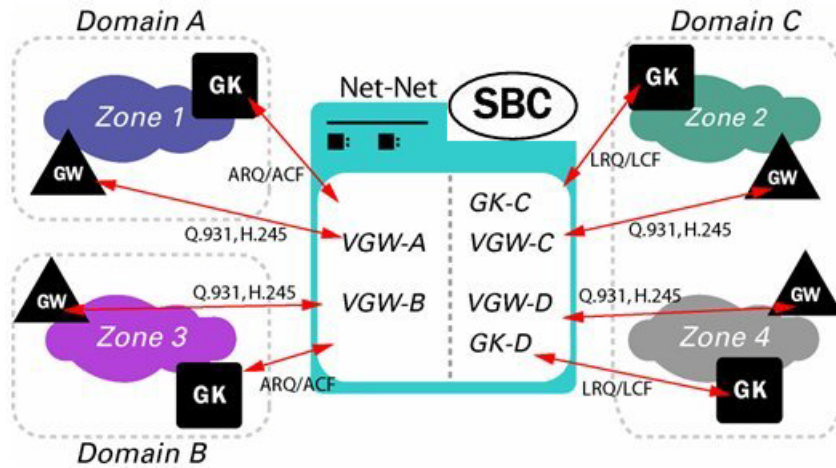
The following diagram illustrates the back-to-back gatekeeper and gateway.



Interworking Gatekeeper/Gateway

In the interworking gatekeeper/gateway signaling mode of operation, the Net-Net SBC interworks between the other two modes; presenting a routed gatekeeper interface to one zone and a gateway to the other.

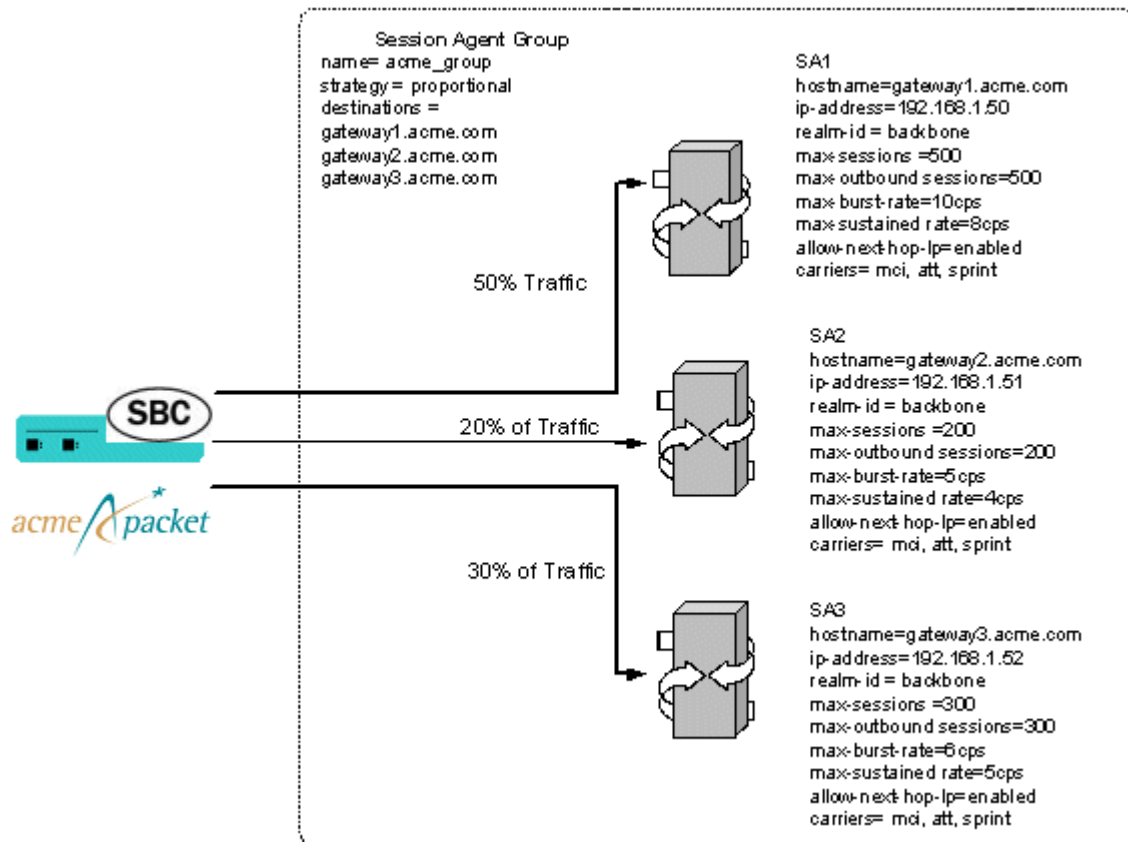
The following diagram illustrates the interworking gatekeeper/gateway.



Load Balancing

This section describes Net-Net SBC load balancing. You can use session agent groups to assist in load balancing among session agents. You define concurrent session capacity and rate attributes for each session agent and then define the session agent group. Next, you select the allocation strategy you want applied to achieve the load balancing you want.

The following example shows a configuration for load balancing gateways based on a proportional allocation strategy.



Routing and load balancing capabilities include the following:

- least cost, which includes cost-based and time-based routing
- customer preference
- traffic aggregation
- routing by media (codec) type
- capacity-based, by destination
- service element load balancing
- service element failure detection and re-route
- session agent failure
- routing by codec

Configuring Routing

This section explains how to configure routing on the Net-Net SBC.

Configuration Prerequisite

You should have already configured the realms for your environment before you configure the routing elements. See *Configuring Realms* for details. You need to know the realm identifier when configuring session agents and local policy.

You can use an asterisk (*) when the session agent exists in multiple realms.

Configuration Order

Recommended order of configuration:

- realm
- media profile
- session agent
- session agent group
- route (local) policy

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by admin@console

Last modified date 2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2.UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Creating a Media Profile

Configure a list of media profiles if you want the local policy to route SIP and H.323 traffic by the codecs specified in the SDP. The list of media profiles entered here are matched against the SDP included in SIP or H.323 requests and the next hop is selected by codec.

The values in this list are matched against the rtpmap attribute of passed SDP and preference weight for route selection is based on the order in which the matching payload type appears in the SDP's media (m=) line.

For example when the following SDP arrives:

```
m=audio 1234 RTP/AVP 0 8 18
```

that contains the following attributes that correspond to three configured local policies with the same cost:

- a=rtpmap:0 PCMU/8000
- a=rtpmap:8 PCMA/8000
- a=rtpmap:18 G729/8000

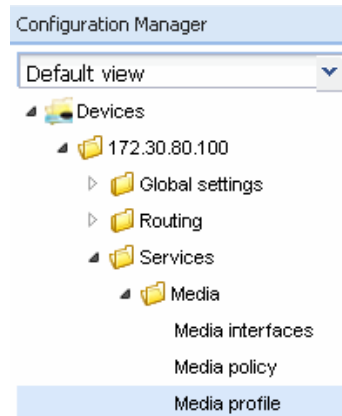
the following route selection action occurs:

The local policy route that corresponds to the a=rtpmap:0 PCMU/8000 attribute is selected because the payload type of 0 in the attribute line matches the first payload type of 0 listed in the m= line. The codec value of PCMU indicated in this selected attribute is used to find the local policy with the media profiles attribute that includes PCMU in the list.

Because the value you enter here is matched against the codec values included in the actual passed SDP, it must correspond to accepted industry-standard codec values.

To create a media profile:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**.

4. **Media profile name**—Select from the combination box.

5. **Subname**—Enter a subname to create multiple media profiles with the same codec name; using a bandwidth value is convenient. For example, you might set a subname of 64k for media profile with **name** value of PCMU
6. Click **OK**. The new media profile appears in the media profile table.
7. Click the new media profile in the media profile table and click **Edit**. The Edit Media profile window appears.
8. **Media type**—Choose the type from the combination box.

Media type

9. **Payload type**—Enter the payload type number. This field is used by the system to determine the encoding type when the SDP included with a session identifies the standard payload type on the m line, but does not include an a-rtpmap entry.
- For RTP/AVP media-profile elements, this field should only be configured when there is a standard payload type number that corresponds to the encoding name. No payload type number is assigned for newer, dynamic codecs.
10. **Transport protocol**—Select the transport protocol from the combination box.
11. **Frames per RTP packet**—Enter the maximum number of frames to use per RTP packet, which facilitates Slow Start translations to Fast Start. Min: 0 / Max: 256. A value of 0 means that this field is not being used.
12. **Bandwidth requirements (Kbps)**—Enter the total bandwidth that the media requires in kilobits per second. Min: 0 / Max: $2^{32}-1$.

13. **Peak rate limit (bytes/sec)**—Enter the peak rate limit in bytes per second. This rate limit is the flowspec parameter r (bucket rate) / p (peak rate) value to insert into COPS message for RACF/PDP configuration.
14. **Maximum burst size (bytes)**—Enter the maximum burst size in bytes. This burst size is the flowspec parameter b (bucket depth) / m (minimum policed unit) / M (maximum datagram size) value to insert into COPS message for RACF/PDP configuration.
15. **Media policing**—Select enabled from the combination box too allow media flow policing.

Media type	audio ▼
Payload type	0
Transport protocol	RTP/AVP ▼
Frames per RTP packet	1
Bandwidth requirements (Kbps)	120
Parameters	
Peak rate limit (bytes per second)	0
Maximum burst size (bytes)	0

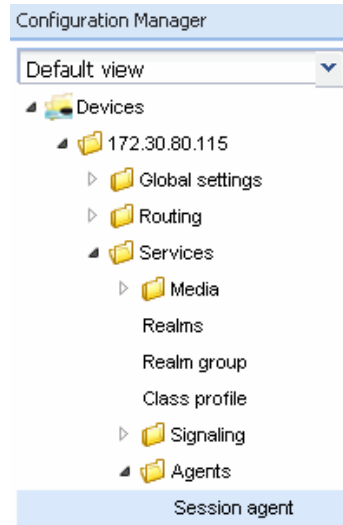
16. **Maximum bandwidth allowed per flow (bytes/sec)**—Enter the sustained rate limit in bytes per second for the media flow, with a maximum of 125Mbytes per second.

Maximum bandwidth allowed per flow (bytes per second),Media policing	3600

17. Click **Apply**.

Creating a Session Agent

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click **Add**.
4. **Name(FQDN)**—Enter the FQDN in the Host name field. If you enter the host name as an IP address, you do not have to enter an IP address in the optional IP address parameter. If you enter the host name in FQDN format, and you want to specify an IP address, enter it in the optional IP address parameter. Otherwise you can leave the IP address parameter blank to allow a DNS query to resolve the host name.

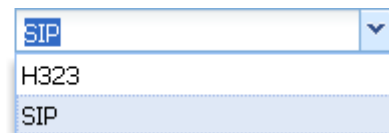
If the initial DNS query for the session agent fails to get back any addresses, the session agent is put out-of-service. When session agent is pinged, the DNS query is repeated. The ping message is not sent until the DNS query gets back one or more IP addresses. After the query receives some addresses, the ping message is sent. The session agent remains out of service until one of the addresses responds.

Note: The value you enter here must be unique to this session agent. No two session agents can have the same hostname.

The hostnames established in the session agent populate the corresponding fields in other elements.

5. Click **Apply**. The new session agent appears in the Session agents table.
6. Select the session agent and click **Edit**.
7. **Protocol**—Select whether this is a SIP or H.323 session agent from the combination box.

Protocol



8. Click **Apply**. The new session agent appears in the Session agents table.

Configuring a SIP Session Agent

To configure a SIP session agent:

1. Select a SIP session agent from the Session agents table and click **Edit**. The Edit Session agent window appears in the content area.
2. **IP Address (required for SIP pings to test SA availability)**—Enter an IP address for the hostname you entered in the Session agent name. You can leave this blank if you want DNS to resolve the name. You must include an IP address if you want to use SIP pings to contact the session agent to confirm whether it is in service.
3. **Description**—Enter a description of session agent.
4. **State**—Set the operational state of this session agent from the combination box.
5. **Realm**—Set the realm where this session agent resides from the combination box.

The realm ID identifies the realm for sessions coming from or going to this session agent. For requests coming from this session agent, the realm ID identifies the ingress realm. For requests being sent to this session agent, the realm ID identifies the egress realm. In a Net-Net SBC, when the ingress and egress realms are different, the media flows must be *steered* between the realms.

- no value: the egress realm is used unless the local policy dictates otherwise
- asterisk (*): keep the egress realm based on the Request URI

6. **IP port**—Enter the port number where this session agent is listening.
 - zero (0): if you enter zero (0), the Net-Net system will not initiate communication with this session agent (although it will accept calls).
 - 1025 through 65535

The default value is 5060.

Note: If the transport method value is TCP, the Net-Net system will initiate communication on that port of the session agent.

7. **Transport protocol**—Set the transport protocol used to communicate with this session agent from the combination box.

IP Protocol	Description
UDP	Each UDP header carries both a source port identifier and destination port identifier, allowing high-level protocols to target specific applications and services among hosts.
UDP+TCP	Allows an initial transport method of UDP, followed by a subsequent transport method of TCP if and when a failure or timeout occurs in response to a UDP INVITE. If this transport method is selected, INVITEs are always sent through UDP as long as a response is received.
DynamicTCP	dTCP indicates that dynamic TCP connections are the transport method for this session agent. A new connection must be established for each session originating from the session agent. This connection is torn down at the end of a session.

IP Protocol	Description
StaticTCP	sTCP indicates that static TCP connections are the transport method for this session agent. Once a connection is established, it remains and is not torn down.
StaticSCTP	sSCTP indicates that static SCTP connections are the transport methods for this session agent. Once a connection is established, it remains and is not torn down.

8. **Privacy header**—Enable this session agent as a trusted source which the Net-Net SBC checks when it receives a message to determine if the source is trusted. Values are:

Proxy Action	Description
none	Session agent is not trusted.
all	Session agent is trusted for all requests and responses.
trust-egress	Session agent is trusted for all egress requests and responses.
trust-ingress	Session agent is trusted for all ingress requests and responses.

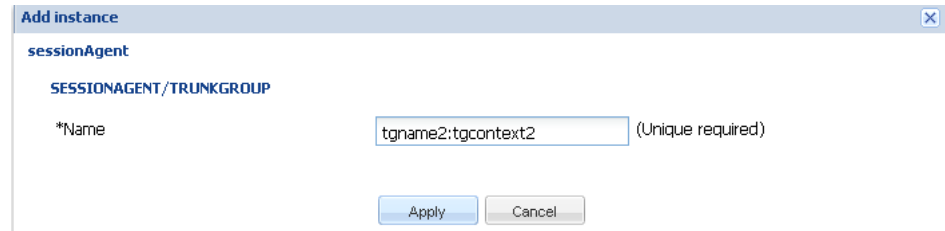
9. **Trusted agent**—Enter a P-Asserted-Identity header that the Net-Net SBC will insert into any initial request for a dialog or standalone transaction sourced by the UE. The inclusion or exclusion of the P-Asserted-Identity header is dependent on the trust level of an egress realm.

*Name(FQDN)	172.43.30.221
IP Address (required for SIP pings to test SA availability)	172.43.30.221
Protocol	SIP
Description	eng lab to NAT
State	enabled
Realm	INSIDE
IP port	5061
Transport protocol	DynamicTLS
Trusted agent	enabled
P-Asserted-identity for IMS	sip:name@acmepacket.com

10. In the Trunk group(s) area, click **Add**. The Add instance dialog box appears.
11. **Trunk group**—Enter the trunk group name and trunk group context in one of the following formats:

tgrp:context

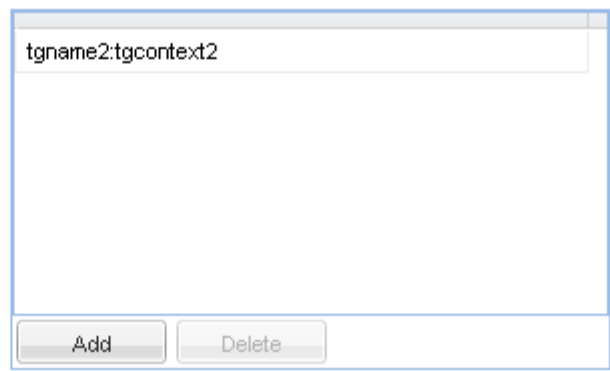
tgrp.context

The image shows a dialog box titled "Add instance" with a close button (X) in the top right corner. Below the title bar, the text "sessionAgent" is displayed. Underneath, "SESSIONAGENT/TRUNKGROUP" is shown in blue. A label "*Name" is followed by a text input field containing "tgrname2:tgcontext2". To the right of the input field, the text "(Unique required)" is displayed. At the bottom of the dialog box, there are two buttons: "Apply" and "Cancel".

Note: If you do not include a context, the Net-Net SBC uses the trunk group context you set in the realm.

12. Click **Apply**. The new entry appears in the Trunk group(s) list. For example:

Trunk group(s)

The image shows a list box containing the text "tgrname2:tgcontext2". Below the list box, there are two buttons: "Add" and "Delete".

SIP Message Handling

1. Scroll to SIP message handling.
2. **Mode for processing requests from SA**—Select the proxy mode behavior from the combination box. Indicate the proxy mode to use when a SIP request arrives from this session agent.

If this field is empty (upon initial runtime or upgrade), its value is set to the value of the SIP configuration's proxy mode by default. If no proxy mode value was entered for the SIP configuration, the default for this field is set to proxy.

The following table lists the available proxy modes along with a brief description.

Proxy Mode	Description
Proxy	If the Net-Net system is a Session Router, the system will proxy the request coming from the session agent and maintain the session and dialog state. If the Net-Net system is a Session Director, the system behaves as a B2BUA when forwarding the request.
Redirect	The system sends a SIP 3xx reDIRECT response with contacts (found in the local policy) to the previous hop.

3. **Action upon receipt of REDIRECT from SA**—Select the redirect action behavior from the combination box. indicate the action you want the SIP proxy to take when it receives a Redirect (3XX) response from the session agent.

If the response comes from a session agent and this field is empty (upon initial runtime or upgrade), the redirect action will be recurse. If no session agent is found (for example, if a message comes from an anonymous user agent), the redirect action is set to proxy. If the Redirect (3xx) response does not have any Contact header, the response will be sent back to the previous hop.

The following table lists the available proxy actions along with a brief description.

Proxy Action	Description
Proxy	The SIP proxy passes the response back to the previous hop; based on the proxy mode of the original request.
Recurse	The SIP proxy serially sends the original request to the list of contacts in the Contact header of the response (in the order in which the contacts are listed in the response). For example, if the first one fails, the request will be sent to the second, and so on until the request succeeds or the last contact in the Contact header has been tried.

4. **SDP included with INVITEs sent to SA**—Set whether this Net-Net SBC should include media session descriptions in Invite or Reinvite messages sent by this session agent from the combination box.
5. Set send media session to enabled if you want to include a media session description (for example, SDP) in the INVITE or REINVITE message sent by the Net-Net system. Setting this field to disabled prevents the Net-Net system from establishing flows for that INVITE message.

Note: Only set send media session to disabled for a session agent that always redirects requests. It returns an error or 3xx response instead of forwarding an INVITE message.

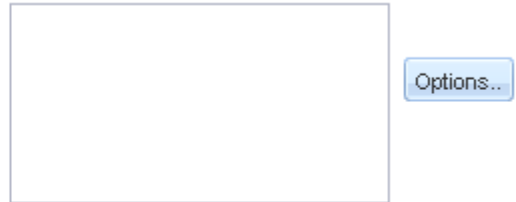
SIP message handling

Mode for processing requests from SA	Proxy	▼
Action upon receipt of REDIRECT from SA	Proxy	▼
SDP included with INVITEs sent to SA	enabled	▼

Options

1. Click **Options** next to the SIP options for this SA area.

SIP options for this SA



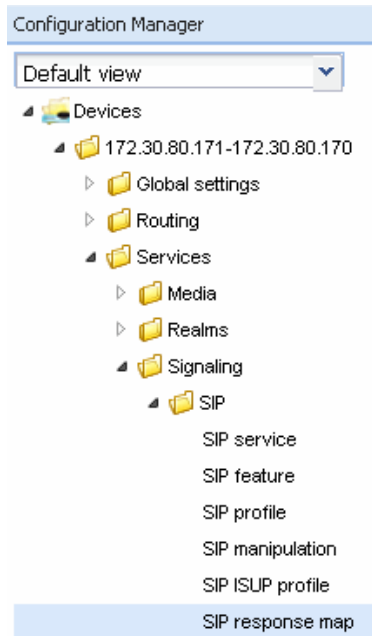
The Options dialog box appears.

2. Click the option name in the Available options list to select it.
3. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
4. Click **OK**.
5. Click **Apply**.

Response Code Translation

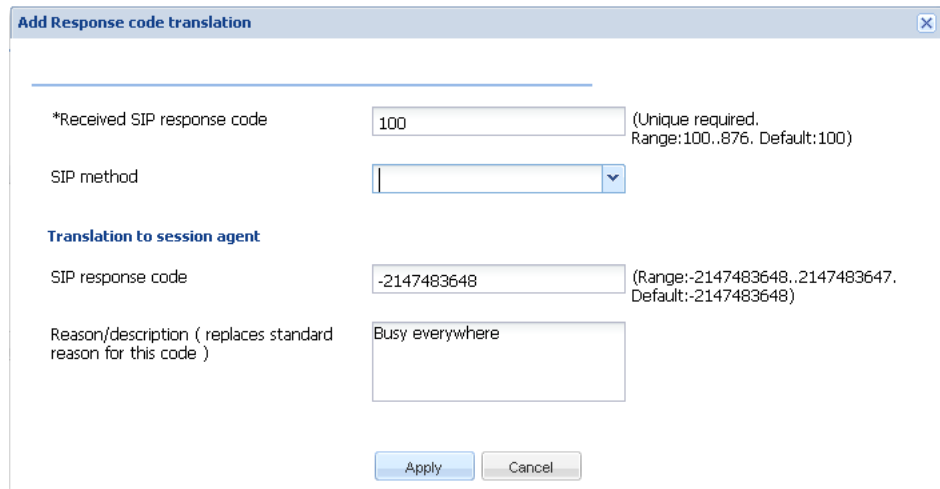
To create a SIP response code map for a session agent:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP response map. The SIP response map table appears in the content area.
3. Click **Add**. The Add SIP response code translation window appears.
4. **SIP response code translation profile Name**—Enter the name for the response code translation profile and click **Apply**. The new profile appears in the Response map table.
5. Click the new response code profile in the table and click **Edit**.
6. Click **Add** in the Response code translation table. The Add SIP response code translation dialog box appears.
7. **Received SIP response code**—Enter the original SIP response code as an integer from 100 to 699.
8. **SIP response code**—Enter the SIP response code into which you want the original response code to be translated. Use an integer from 100 to 699.

9. **Reason/description (replaces standard reason for this code)**—Enter a reason for the translated code. This response comment is sent with the translated code.



Add Response code translation

*Received SIP response code: (Unique required. Range:100..876. Default:100)

SIP method:

Translation to session agent

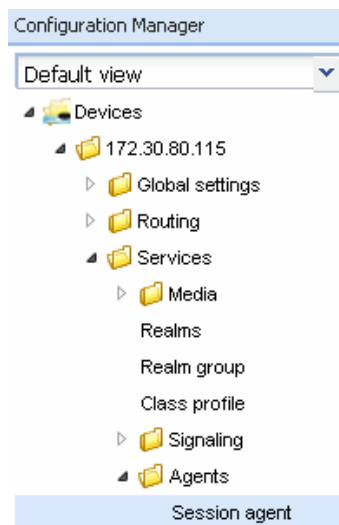
SIP response code: (Range:-2147483648..2147483647. Default:-2147483648)

Reason/description (replaces standard reason for this code) :

10. Click **Apply**.

To assign a response code map

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent you want to configure in the table and click **Edit**.
4. Scroll to the SIP message handling section.
5. **Local response code translation profile**—Choose the local response code translation profile from the combination box.

Local response code translation profile

6. **User part to be used in PING To header**—Enter the user in the User part to be used in PING To header.

7. **User part to be used in PING From header**—Enter the user and user URI portion in the User part to be used in PING From header.

User part to be used in PING To header

User part to be used in PING From header

8. Click **Apply**.
9. Click on the Translation and manipulation tab at the top of the content area.
10. **Manipulation ruleset for inbound messages**—Select the SIP header manipulation ruleset you want to apply to SIP packets in inbound messages.
11. **Manipulation ruleset for outbound messages**—Select the SIP header manipulation ruleset you want to apply to SIP packets in outbound messages.

Address translations attached to session agents take precedence over address translations attached to realms. If no address translation is applied to a session agent, then the Net-Net SBC will use the address translation applied to a realm. If an address translation is applied to both a realm and session agent, the translation attached to the session agent will apply. If the applicable session agent and realm have no associated translations, then the addresses will remain in their original forms and no address translations will be performed.

Manipulation ruleset

Manipulation ruleset for inbound messages

Manipulation ruleset for outbound messages

12. **Trust me for LI**—Select enabled from the combination box if this session agent will be trusted for lawful intercept (LI).

Trust me for LI

13. Click **Apply**.

SIP Session Routing and Address Translation

To enter information in the SIP session agent Translation and manipulation tab:

1. In the session agent data window, scroll to State.
2. **Loose routing**—Selected enabled from the combination box.

When the SIP NAT route home proxy parameter is enabled, the Net-Net SBC looks for a session agent that matches the home proxy address and checks the loose routing value. If loose routing is enabled, a Route header is included in the outgoing request in accordance with RFC 3261. If loose routing is disabled, the Route header is not included in the outgoing request (in accordance with strict routing procedures defined in RFC 2543).

The loose routing value is also checked when the local policy's next hop value matches a session agent. If loose routing is set to enabled, the outgoing request retains the original Request-URI and Route header with the next hop address.

Loose routing

3. Click the Translation and manipulation tab at the top of the content area.
4. **Incoming embedded headers to be converted to non-embedded headers**—Click Add.

Incoming embedded headers to be converted to non-embedded headers

The Add instance dialog box appears.

5. **Embedded headers**—Enter the name of the header you want to embed. When the specified embedded header is found, the full <header=value> pair is inserted as a unique header in a redirected INVITE message that exits the Net-Net SBC.
6. Click Apply.

The header name appears in the Embedded headers list.

7. **Include Carrier as**—Select the method from the combination box.

You can set this parameter to let the system perform simple digit translation on calls sent to gateways. A 3-digit prefix is inserted in front of the telephone number (the Request-URI) that the gateway will use to select a trunk group. Most often, the Net-Net system needs to insert the carrier code into the signaling message that it sends on.

Include Carrier As

The following table lists the available modes..

Carrier Mode	Description
none	Default value. Carrier information will not be added to the outgoing message.

Carrier Mode	Description
URI parameter	Adds a parameter to the Request-URI. For example, cic-XXX.
telephone number prefix	Adds the carrier code as a prefix to the telephone number in the Request-URI (in the same manner as PSTN).

8. **Inbound**—Select a profile from the Inbound combination box in the Address translation profiles area..

Address translation profiles

Inbound		▼
Outbound		▼

9. **Ping method**—Select the Ping method from the combination box. The ping confirms whether the session agent is in service. If this field is left empty, no session agent will be pinged.
- Setting this field value to the OPTIONS method might produce a lengthy response from certain session agents and could potentially cause performance degradation on your Net-Net system.
10. **Ping interval (sec)**—Enter the interval in seconds. The Net-Net SBC only sends the ping if no SIP transactions (have occurred to/from the session agent within the time period you enter here.

Ping method	INFO	▼
Ping interval(sec)	32	

11. Click **Apply**.

SIP Session Agent Constraints

To enter information in the SIP session agent constraints tab:

- Click the Constraints tab at the top of the content area.
- Session constraints**—Select enabled from the combination box to activate SIP session agent constraints.
- Max. number of sessions (#)**—Enter the maximum number of inbound and outbound sessions.
 - minimum: zero (0) is the default value and means there is no limit
 - maximum: $2^{32} - 1$
- Max. INVITE burst rate (# sessions per sec)**—Enter the maximum SIP INVITE burst rate in sessions per second. This is the number of session invitations allowed to be sent to or received from the session agent within the configured burst rate window value.

SIP session invitations arrive at and leave from the session agent in intermittent bursts. By entering a value in this field, you can limit the amount of session invitations that are allowed to arrive at and leave from the session agent.

For example, if you enter a value of 50 here and a value of 60 (seconds) for the burst rate window constraint, no more than 50 session invitations can arrive at

or leave from the session agent in that 60 second time frame (window). Within that 60-second window, any sessions over the limit of 50 are rejected.

- minimum: zero (0) session invitations per second
- maximum: $2^{32} - 1$ session invitations per second

5. **Max. INVITE sustained rate (# sessions per sec)**—Enter the maximum SIP INVITE sustained rate in sessions per second. This is the maximum rate of session invitations allowed to be sent to or received from the session agent within the current window. The value you set here must be larger than the value you set for the maximum burst rate constraint.

- minimum: zero (0) invitations per seconds
- maximum: $2^{32} - 1$ invitations per seconds

The current rate is determined by counting the number of session invitations processed within a configured time period and dividing that number by the time period. By entering a value in this field, you can limit the amount of session invitations that are allowed to arrive at and leave from the session agent over a sustained period of time.

For the sustained rate, the Net-Net system maintains a current and previous window size. The period of time over which the rate is calculated is always between one and two window sizes.

6. **Max. inbound sessions (#)**—Enter the maximum inbound sessions allowed for this constraint. The minimum value is 0 (default), and the maximum is 999999999.
7. **Max. inbound burst rate (# sessions per sec)**—Enter the maximum inbound burst rate (number of session invitations per second) for this constraint. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
8. **Max. inbound sustained rate (# sessions per sec)**—Enter the maximum inbound sustain rate (of session invitations allowed within the current window) for this constraint. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
For the sustained rate, the Net-Net SBC maintains a current and previous window size. The period of time over which the rate is calculated is always between one and two window sizes.
9. **Max. outbound sessions (#)**—Enter the maximum outbound sessions allowed for this constraint. The minimum value is 0 (default), and the maximum is 999999999.
10. **Max. outbound burst rate (# sessions per sec)**—Enter the maximum outbound burst rate (number of session invitations per second) for this constraint. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
11. **Max. outbound sustained rate (# sessions per sec)**—Enter the maximum outbound sustain rate (of session invitations allowed within the current window) for this constraint. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
12. **Burst window (sec)**—Enter the time in seconds that you want to use to measure the burst rate; the “window” is the time over which the burst rate is calculated, and is used for the over all burst rate as well as the inbound and outbound burst rates. The minimum value for this parameter is 0 (default), and the maximum is 999999999.

13. **Sustained window**—Enter the time in seconds used to measure the sustained rate; the “window” is the time over which the sustained rate is calculated, and is used for the over all sustained rate as well as the inbound and outbound sustained rates. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
14. **Max. rate of registration messages sent to SA (reg/sec)**—Enter the time in seconds after which the SIP proxy should resume sending session invitations to the session agent. This value only takes effect when the SIP proxy stops sending invitations because a constraint is exceeded.

- minimum: zero (0) seconds
- maximum: $2^{32} - 1$ seconds

Session constraints	enabled 
Max. number of sessions (#)	100
Max. INVITE burst rate (# sessions per sec)	
Max. INVITE sustained rate (# sessions per sec)	0
Max. inbound sessions (#)	0
Max. inbound burst rate (# sessions per sec)	0
Max. inbound sustain rate (# sessions per sec)	0
Max. outbound sessions (#)	0
Max. outbound burst rate (# sessions per sec)	0
Max. outbound sustain rate (# sessions per sec)	0
Burst rate window (sec)	0
Sustained rate window (sec)	0
Max. rate of registration messages sent to SA (reg/sec)	0

15. Scroll to the Answer seizure ratio section of the content area.
16. **Minimum number of seizures**—Enter the minimum number of seizures that when exceeded, cause the session agent to be marked as having exceeded its constraints. Calls will not be routed to the session agent until the time-to-resume has elapsed. The range of values is 1 to 999999999.
17. **Minimum answer to seizure ratio percentage**—Enter the percentage you want as the minimum answer to seizure response. If the session agent’s ASR for the current window falls below this percentage, the session agent is marked as

having exceeded its constraints and calls will not be routed to it until the time-to-resume has elapsed.

Answer seizure ratio

Minimum number of seizures	5
Minimum answer to seizure ratio percentage	0

18. Scroll Time-to-wait timers.

SIP Timers

1. **After threshold reached before resuming INVITES to SA**—Enter the time in seconds the SIP proxy must wait between the time an invitation is sent to an SA and the time it tries again.
2. **After SA unresponsive before resuming INVITES to SA**—Enter the maximum number of registration messages per second you want sent to this session agent.
3. **After communication established before SA considered 'in service'**—Enter the time in seconds the session agent must be operational before it is considered in-service.

Time-to-wait timers (sec)

After threshold reached before resuming INVITES to SA	100
After SA unresponsive before resuming INVITES to SA	0
After communication established before SA considered 'in service'	30

Response Codes for Recursion

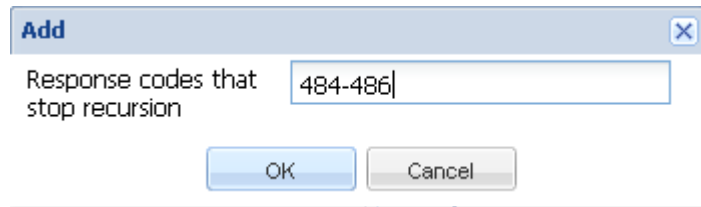
1. List of response codes that stop recursion—Click Add.

Response codes that stop recursion

AddDelete

The SIP Response codes dialog box appears.

2. **SIP response code**—Enter the response code value. You can enter a single response code or response code range; or a comma-delimited list of response codes or response code ranges.



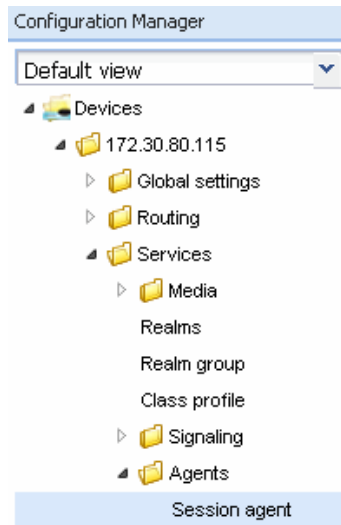
A dialog box titled "Add" with a close button (X) in the top right corner. It contains a text input field with the label "Response codes that stop recursion" and the value "484-486". Below the input field are two buttons: "OK" and "Cancel".

3. Click **OK** to close the dialog box. The response code value appears in the list.
4. Click **Apply**.

Configuring an H.323 Session Agent

To configure an H.323 session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click **Add**.
4. **Name(FQDN)**—Enter the FQDN in the Host name field. If you enter the host name as an IP address, you do not have to enter an IP address in the optional IP address parameter. If you enter the host name in FQDN format, and you want to specify an IP address, enter it in the optional IP address parameter. Otherwise you can leave the IP address parameter blank to allow a DNS query to resolve the host name.

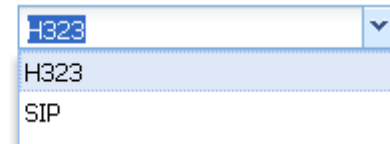
If the initial DNS query for the session agent fails to get back any addresses, the session agent is put out-of-service. When session agent is pinged, the DNS query is repeated. The ping message is not sent until the DNS query gets back one or more IP addresses. After the query receives some addresses, the ping message is sent. The session agent remains out of service until one of the addresses responds.

Note: The value you enter here must be unique to this session agent. No two session agents can have the same hostname.

The hostnames established in the session agent populate the corresponding fields in other elements.

5. Click **Apply**. The new session agent appears in the Session agents table.
6. Select the session agent and click **Edit**.
7. **Protocol**—Select whether this is a SIP or H.323 session agent from the combination box.

Protocol



8. Click **Apply**. The new session agent appears in the Session agents table.
9. **IP Address (required for SIP pings to test SA availability)**—Enter an IP address if you want to ping the session agent to confirm whether it is in service.
10. **Protocol**—Select H323 from the combination box.
11. **Description**—Enter a description of the session agent.
12. **State**—Set the operational state of this session agent from the combination box.
13. **Realm**—Set the realm where this session agent resides from the combination box.
14. **IP port**—Enter the port number where this session agent is listening.
15. **Transport protocol**—Set the protocol used to communicate with this session agent from the combination box.

IP Protocol	Description
UDP	Each UDP header carries both a source port identifier and destination port identifier, allowing high-level protocols to target specific applications and services among hosts.
UDP+TCP	Allows an initial transport method of UDP, followed by a subsequent transport method of TCP if and when a failure or timeout occurs in response to a UDP INVITE. If this transport method is selected, INVITEs are always sent through UDP as long as a response is received.
DynamicTCP	dTCP indicates that dynamic TCP connections are the transport method for this session agent. A new connection must be established for each session originating from the session agent. This connection is torn down at the end of a session.
StaticTCP	sTCP indicates that static TCP connections are the transport method for this session agent. Once a connection is established, it remains and is not torn down.

*Name(FQDN)	HSessionAgent
IP Address (required for SIP pings to test SA availability)	172.43.30.221
Protocol	H323 ▼
Description	lab to NAT
State	enabled ▼
Realm	SLB-INSIDE ▼
IP port	5060
Transport protocol	UDP ▼
Trusted agent	enabled ▼
P-Asserted-identity for IMS	h323:name@acmepacket.com

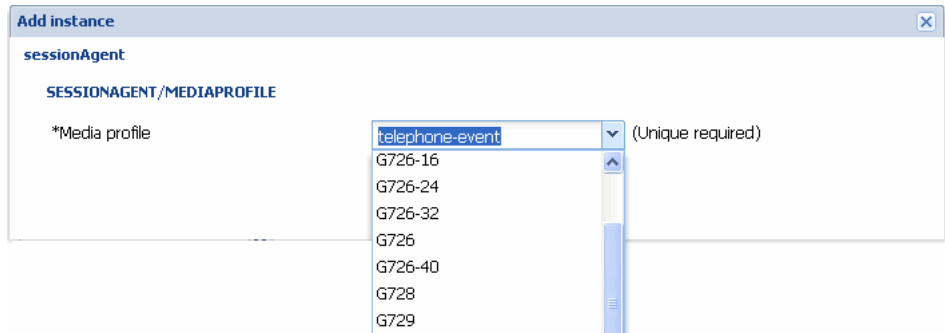
Media Profiles

- 1. List of media profiles—Click Add.

List of media profiles

The Add instance dialog box appears.

2. **Media Profile**—Select the profile from the combination box

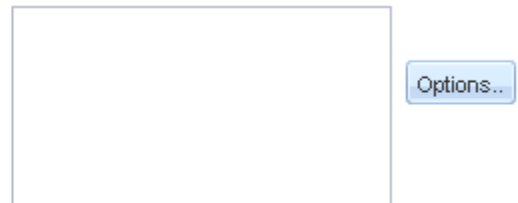


3. Click **Apply**. The media profile name appears in the List of media profiles table.

Options

1. Click **Options** next to the SIP options for this SA area.

SIP options for this SA



The Options dialog box appears.

2. Click the option name in the Available options list to select it.
3. Click  to move the option to the Selected options list. The option name appears along with a description in the Description section of the window.
4. Click **OK**.
5. Click **Apply**.

Routing and Address Translation

To enter information in the H.323 session agent Translation and manipulation tab:

1. **State**—Select the next-hop behavior for this session agent from the combination box.

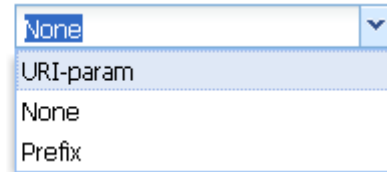
State enabled ▼

2. Click the Translation and manipulation tab at the top of the content area.
3. **Include Carrier as**—Select the method from the combination box.

You can set this parameter to let the system perform simple digit translation on calls sent to gateways. A 3-digit prefix is inserted in front of the telephone number (the Request-URI) that the gateway will use to select a trunk group.

Most often, the Net-Net system needs to insert the carrier code into the signaling message that it sends on.

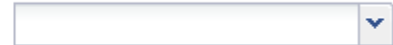
Include Carrier As



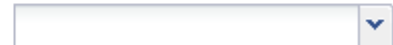
4. **Inbound**—Select a profile from the Inbound combination box in the Address translation profiles area..

Address translation profiles

Inbound



Outbound



5. Click **Apply**.

Session Agent Constraints

To enter information in the H.323 session agent constraints tab:

1. Click the Constraints tab at the top of the content area.
2. **Session constraints**—Select enabled from the combination box to activate H.323 session agent constraints.
3. **Max. number of sessions (#)**—Enter the maximum number of inbound and outbound sessions.
 - minimum: zero (0) is the default value and means there is no limit
 - maximum: $2^{32} - 1$
4. **Max. INVITE burst rate (# sessions per sec)**—Enter the maximum H.323 INVITE burst rate in sessions per second. This is the number of session invitations allowed to be sent to or received from the session agent within the configured burst rate window value.

H.323 session invitations arrive at and leave from the session agent in intermittent bursts. By entering a value in this field, you can limit the amount of session invitations that are allowed to arrive at and leave from the session agent.

For example, if you enter a value of 50 here and a value of 60 (seconds) for the burst rate window constraint, no more than 50 session invitations can arrive at or leave from the session agent in that 60 second time frame (window). Within that 60-second window, any sessions over the limit of 50 are rejected.

 - minimum: zero (0) session invitations per second
 - maximum: $2^{32} - 1$ session invitations per second
5. **Max. INVITE sustained rate (# sessions per sec)**—Enter the maximum H.323 INVITE sustained rate in sessions per second. This is the maximum rate of session invitations allowed to be sent to or received from the session agent within the current window. The value you set here must be larger than the value you set for the maximum burst rate constraint.
 - minimum: zero (0) invitations per seconds
 - maximum: $2^{32} - 1$ invitations per seconds

The current rate is determined by counting the number of session invitations processed within a configured time period and dividing that number by the time period. By entering a value in this field, you can limit the amount of session invitations that are allowed to arrive at and leave from the session agent over a sustained period of time.

For the sustained rate, the Net-Net system maintains a current and previous window size. The period of time over which the rate is calculated is always between one and two window sizes.

6. **Max. inbound sessions (#)**—Enter the maximum number of simultaneous inbound sessions.

- minimum: zero (0) is the default value and means there is no limit
- maximum: $2^{32} - 1$

The value you enter here cannot be larger than the maximum sessions value.

7. **Max. outbound sessions (#)**—Enter the maximum number of simultaneous outbound sessions.

- minimum: zero (0) is the default value and means there is no limit
- maximum: $2^{32} - 1$

The value you enter here cannot be larger than the maximum sessions value.

8. **Burst rate window (sec)**—Enter a new burst rate window in seconds. The term *window* refers to the period of time over which the burst rate is computed. Refer to the maximum burst rate information.

The value you set here must be smaller than the value you set for the maximum burst rate constraint.

For example, if you enter a value of 5000 here and a value of 3600 (seconds) for the sustain rate window constraint, no more than 5000 session invitations can arrive at or leave from the session agent in any given 3600 second time frame (window). Within that 3600-second window, sessions over the 5000 limit are rejected.

9. **Sustained rate window (sec)**—Enter a new sustained rate window in seconds. The value you set here must be larger than the value you set for the maximum sustain rate constraint.

- minimum: zero (0) seconds
- maximum: $2^{32} - 1$ seconds

10. **Max. rate of registration messages sent to SA (reg/sec)**—Enter the time in seconds after which the SIP proxy should resume sending session invitations to the session agent. This value only takes effect when the SIP proxy stops sending invitations because a constraint is exceeded.

- minimum: zero (0) seconds

Session constraints	enabled ▼
Max. number of sessions (#)	100
Max. INVITE burst rate (# sessions per sec)	
Max. INVITE sustained rate (# sessions per sec)	0
Max. inbound sessions (#)	0
Max. inbound burst rate (# sessions per sec)	0
Max. inbound sustain rate (# sessions per sec)	0
Max. outbound sessions (#)	0
Max. outbound burst rate (# sessions per sec)	0
Max. outbound sustain rate (# sessions per sec)	0
Burst rate window (sec)	0
Sustained rate window (sec)	0
Max. rate of registration messages sent to SA (reg/sec)	0
Maximum register burst rate(reg/sec)	0
Register burst window (sec)	0

H.323 Timers

1. **After threshold reached before resuming SETUPS to SA**—Enter the maximum number of registration messages per second you want sent to this session agent.
2. **After SA unresponsive before resuming SETUPS to SA**—Enter the maximum number of registration messages per second you want sent to this session agent.

3. **After communication established before SA considered 'in service'**—Enter the time in seconds the session agent must be operational before it is considered in-service..

Time-to-wait timers (sec)

After threshold reached before resuming
INVITES to SA

After SA unresponsive before resuming
INVITES to SA

After communication established before
SA considered 'in service'

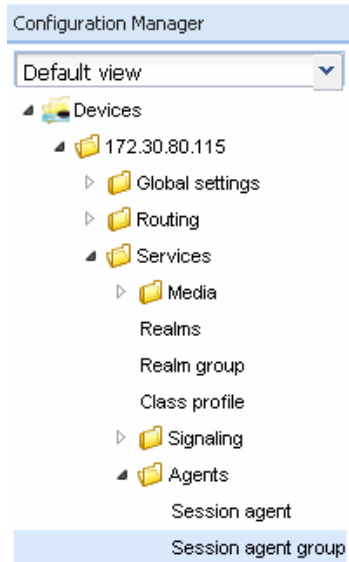
4. Click **Apply**.

Configuring Session Agent Groups

This section describes how to configure session agent groups on your Net-Net SBC from the Net-Net Central.

To configure session agent groups:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent group. The Session agent group table appears in the content area.
3. Click **Add**. The Add instance dialog box appears.
4. **Group name**—Enter the name of this session agent group.
5. Click **Apply**. The new session agent group now appears in the table.
6. Select the session agent group from the table and click **Edit**.
7. **Protocol**—Select the application protocol for this session agent group from the combination box.
8. **Description**—Enter a description for the session agent.

9. **Session agent selection strategy**—Set the selection strategy from the combination box. Indicate the session agent allocation strategy you want to use.

sessionAgentGroup

State	enabled ▼
*Group name	SAGroup1
Protocol	SIP ▼
Description	
Session agent selection strategy	Hunt ▼

The strategy you chose selects the session agents that will be made available by this session agent group. Available values include:.

Allocation Strategy	Description
hunt	Selects session agents in the order in which they are listed. For example, if the first agent is online, working, and has not exceeded defined constraints; all traffic is sent to the first agent. If the first agent is offline or if it exceeds a defined constraint, the second agent is selected. If the first and second agents are offline or exceed defined constraints, the third agent is selected. And so on through the list of session agents.
round robin	Selects each session agent in the order in which they are listed in the destination list, selecting each agent in turn, one per session.
least busy	Selects the session agent that has the fewest number of sessions relative to the maximum outbound sessions constraint or the maximum sessions constraint (for example, lowest percent busy).
proportional distribution	Based on programmed, constrained session limits, the Proportional Distribution strategy proportionally distributes the traffic among all of the available session agents.
lowest sustained rate	The Lowest Sustained Rate strategy routes to the session agent with the lowest sustained rate of session initiations/invitations (based on observed sustained session request rates).

10. Scroll to the Session agents in the group section and click **Add**.

Session agents in this group

Sequence number	Name

Buttons: Add, Delete, Re-order

The Add instance dialog box appears in the content area.

11. **Session agent**—Select a session agent to add to this session agent group by selecting it from the combination box.

Add instance

sessionAgentGroup

SESSIONAGENTGROUP/SIPDEST

*Name: 1.1.1.1 (Unique required)

seqno

Buttons: Apply, Cancel

12. Click **Apply**.
13. Repeat this process to add multiple session agents to the session agent group. The session agent names appear in the table.

Trunk Groups (SIP only)

1. **Trunk group(s)**—Click **Add**.

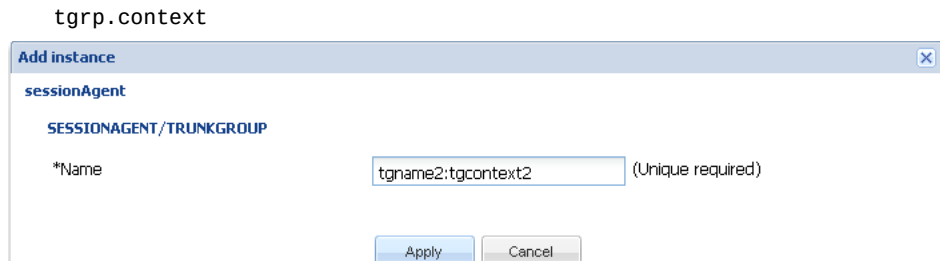
Trunk group(s)

tgname2tgcontext2

Buttons: Add, Delete

The Add instance dialog box appears.

2. **Trunk group**—Enter the trunk group name and trunk group context in one of the following formats:
- tgpr:context



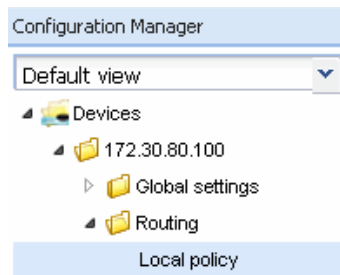
Note: If you do not include a context, the Net-Net SBC uses the trunk group context you set in the realm.

3. Click **Apply**. The new entry appears in the Trunk group(s) list. For example:

Configuring Routes

This section describes how to configure routes/local policy using the Net-Net Central.

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.

Creating Routes

1. Click **Add** in the Local policy table. The Add instance dialog box appears.
2. **To Address**—Enter a To Address value. You can use the asterisk (*) as a wildcard to indicate all this policy can be used for any destination address.

You can also use E.164 addresses (strings that contain telephone keypad characters) here. Number matching works from left to right. Formats include the following:

- SIP Request-URI
- FQDNs
- IP addresses
- H.323 CalledPartyAddress.

3. **From Address**—Enter a From Address value. You can use the asterisk (*) as a wildcard to indicate this policy can be used with all originating addresses.

You can also use complete or partial E.164 addresses (strings that contain telephone keypad characters) here. Number matching works from left to right. Formats include the following:

- SIP From address
- FQDNs
- IP addresses

- H.323 CallingPartyAddress
4. **Source realm**—Set the Source realm from the combination box to determine how to route traffic. This list identifies from what realm traffic is coming and is used for routing by ingress realm by the local policy.

You can use the asterisk (*) as a wildcard to indicate this local policy can be used with all realms. Or you can enter a value that corresponds to the identifier of an already configured realm. Formats include the following:

- realm ID
- customer name
- peer name
- subdomain name
- VPN identifier

5. Click **Apply**. The Local policy route appears in the table.

6. Select the Local policy route from the table and click **Edit**.
7. **State**—Set the state of the local policy from the combination box to indicate whether you want to enable or disable the local policy.
8. **Priority**—Set the minimum priority to forward an anonymous message from the combination box.
9. **Activate time**—Enter the time you want the local policy to be activated using the following syntax:

yyyy-mm-dd hh:mm:ss

The default value of 0000 implies that the defined policy attributes can be considered in effect any time after 00:00:00.

10. **Deactivate time**—Enter the time you want the local policy to be deactivated using the following syntax:

yyyy:mm:dd hh:mm:ss

The default value of 2400 implies that the defined policy attributes can be considered in effect any time before midnight.

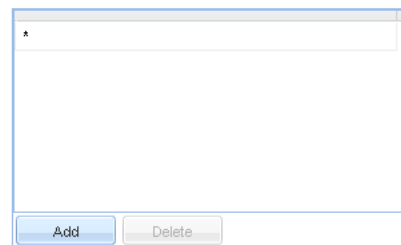
11. **Description**—Enter an optional descriptive string to identify the configuration and its function.

State	enabled ▼
Priority	none ▼
Activate time	2011-03-31 20:30:00
Deactivate time	
Description	Sample description

12. In the From area, click **Add**.

From

(Number/address/host)



The dialog box shows a text input field containing an asterisk (*). Below the input field are two buttons: 'Add' and 'Delete'.

The Add instance dialog box appears.

13. **Address**—Enter the originating address information by entering a From address value. You can use the asterisk (*) as a wildcard to indicate this policy can be used with all originating addresses. You can also use complete or partial E.164 addresses here. Number matching works from left to right. Formats include the following:

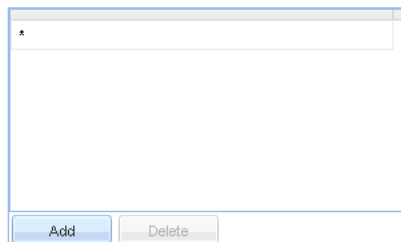
- SIP From address
- FQDNs
- IP addresses
- H.323 CallingPartyAddress

14. Click **Apply**. The new From address appears in the table.

15. In the Request URI area, click **Add**.

Request URI

(Number/address/host)



The dialog box shows a text input field containing an asterisk (*). Below the input field are two buttons: 'Add' and 'Delete'.

The Add instance dialog box appears.

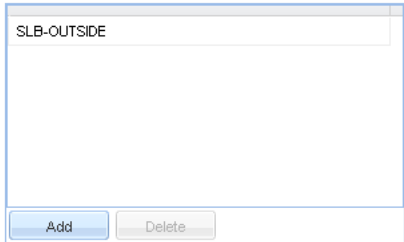
16. **Address**—Enter the destination address by entering a To address value. You can use the asterisk (*) as a wildcard to indicate all this policy can be used for

any destination address. You can also use E.164 addresses here. Number matching works from left to right. Formats include the following:

- SIP Request-URI
- FQDNs
- IP addresses
- H.323 CalledPartyAddress

17. Click **Apply**. The new Request URI address appears in the Request URI list.
18. In the Source realms area, click **Add**.

Source realms



The Add instance dialog box appears.

19. **Source realm**—Choose a realm from the combination box. This list identifies from what realm traffic is coming and is used for routing by ingress realm by the local policy.
20. Click **Apply**. The new realm name appears in the Source realm list.
21. Click **Apply** to save the route information.

Configuring Route Policies

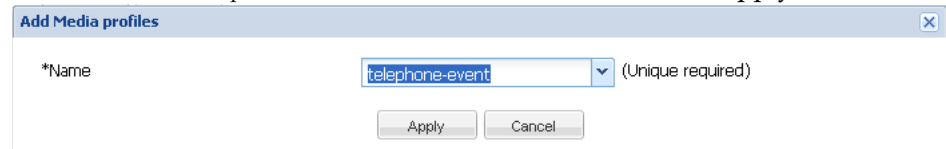
22. Scroll to the Route policy attributes table and click Add.
23. **Start time**—Set the time of day these policy attributes considered for preference determination.
24. **End time**—Set the time of day these policy attributes cease to be considered for preference determination.
25. **Days of the week**—Enter the combination of days of the week plus holidays that policy attributes can be considered for preference determination. A holiday entry coincides with a configured holiday. At least one day or holiday must be specified in this field.
26. **Next hop**—Set the Next hop from the combination box. You can use the following as next hops:
 - IPv4 address of a specific endpoint
 - Hostname or IPv4 address of a configured session agent
 - Group name of a configured session agent group
 - 0.0.0.0 (if used for a next hop the request is terminated and the Net-Net SBC sends the response 404 Not Found)
27. **Action**—Set this parameter to redirect if you want to send a redirect next-hop message back to the calling party with the information in the Contact. The calling party then needs to send an INVITE using that information.
28. **Terminate recursion**—Select enabled from the combination box to terminate route recursion with this next hop.

29. **Egress protocol**—Select the protocol from the combination box that will be used when sending messages to the next hop.
30. Click **Apply**.

Add Route policy attributes	
*Start time	0000
*End time	2400
*Days of the week	U-S
*Next hop	
*Action	none
*Terminate recursion	disabled
*Egress protocol	
Apply Cancel	

31. Click **Apply**. The route appears in the Route table.
32. Select the Route policy and click Edit.
33. **State**—Select enabled from the combination box to enable this route policy.
34. **Cost**—Enter a cost value that acts as a unitless representation of the cost of a route relative to other routes reaching the same destination (To address) in the cost textbox. This value is used as a way of ranking policy attributes.
 - minimum: zero (0)
 - maximum: 999999999
35. Click **Add** in the Media profiles table. The Add Media profiles dialog box appears.

36. Select a media profile from the combination box and click **Apply**.



The dialog box titled "Add Media profiles" contains a label "*Name" followed by a dropdown menu showing "telephone-event" and the text "(Unique required)". At the bottom are "Apply" and "Cancel" buttons.

37. Click **Apply**.

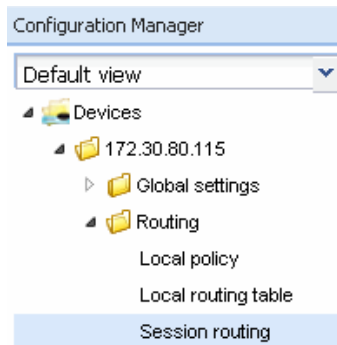
38. Click **Back** to return to the Local policy data screen.

Configuring Holidays

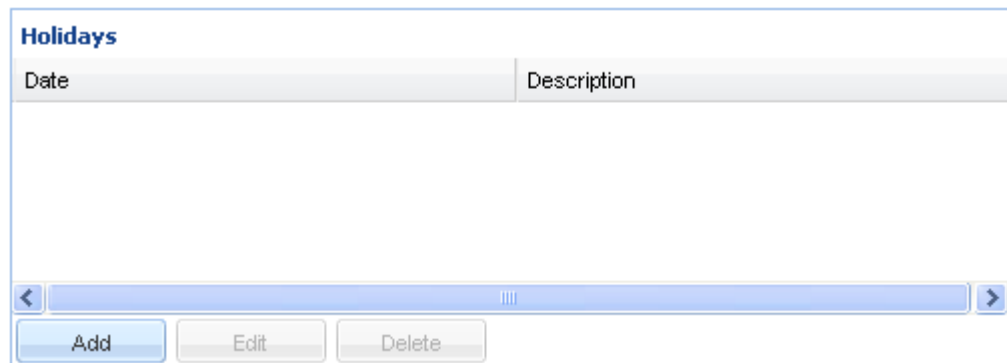
You can configure the holidays for which you want the route policy attributes to be in effect using the Net-Net Central. A holiday entry must correspond with a configured holiday established in the session router.

To configure holidays:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. Click **Add** in the Holiday dates and descriptions section..



The "Holidays" section displays a table with two columns: "Date" and "Description". Below the table is a scrollable area with a search bar and three buttons: "Add", "Edit", and "Delete".

The Add instance dialog box appears.

5. **Date**—Enter the date of the holiday in the Date textbox.
 - Enter value in the format yyyy-mm-dd
6. **Description**—Add descriptive text in the Description textbox.

The screenshot shows a dialog box titled "Add instance" with a close button (X) in the top right corner. Below the title bar, the text "sessionRouter" is displayed. Underneath, "SESSIONROUTER/HOLIDAY" is shown. There are two input fields: one for "*Description" containing "2006-12-25" and another for "*Date" containing "Christmas". Both fields have "(Unique required)" text to their right. A "Configuration Guidelines" link with a dropdown arrow is located to the right of the date field. At the bottom of the dialog are "Apply" and "Cancel" buttons.

7. Click **Apply**. The holiday information appears in the list.
8. Click **Apply**.

Route (Local) Policy Matching for Parent Realms

For SIP and H.323, you can configure the Net-Net SBC to use the parent realm for routing purposes even when the source realm for an incoming message is a child realm.

With this feature disabled (default), the Net-Net SBC uses the specific source realm to perform a route policy look-up. When the source realm is a child realm and any relevant local policies are configured with the parent realm, there will be no matches and the route policy look-up will fail. To avoid this issue and ensure successful look-ups, you must configure multiple local policies if you want to use a configuration with nested realms.

The Net-Net SBC examines the source realm to determine if it is a parent realm with any child realms when you enable this feature. If the parent, source realm does have child realms, then the Net-Net SBC creates route policy entries for the parent and all of its child realms. This operation is transparent and can save time during the configuration process.

It is possible, then, for a route policy look-up to match the same child realm in two ways:

- Through a match via the parent realm
- Through a direct match for a route policy configured with that specific child realm

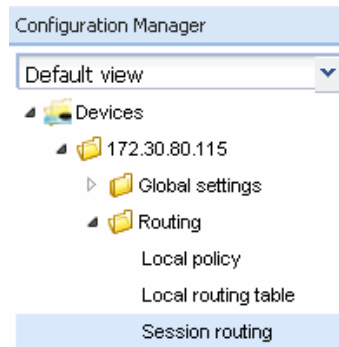
In such a case, the child realm must have different costs for each type of match to avoid collisions.

Configuring Route (Local) Policy Matching

This feature is enabled on a global basis in the session router configuration. Because it applies system-wide, all source realms will use this form of matching when enabled.

To enable route policy matching for parent realms:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. **Include parent realm in route policy matching**—Select enable from the combination box if you want the Net-Net SBC to perform route policy realm matching based on the parent realm (so that there are route policy entries for parent and child realms).

Include parent realm in route policy matching

5. Click **Apply**.

Answer to Seizure Ratio-Based Routing

SIP and H.323 session agent constraints set a threshold for Answer to Seizure Ratio (ASR) has been implemented. ASR is considered when determining whether session agents are within their constraints to route calls (in addition to session and rate constraints).

The new session agent constraints indicate the minimum acceptable ASR value and computes the ASR while making routing decisions. ASR is calculated by taking the number of successfully answered calls and dividing by the total number of calls attempted (which are known as seizures).

If the ASR constraints are exceeded, the session agent goes out of service for a configurable period of time and all traffic is routed to a secondary route defined in the local policy (next hop with higher cost).

How It Works

The two session agent constraints are:

- minimum seizure: determines if the session agent is within its constraints. When the first call is made to the session agent or the if calls to the session agent are not answered, the minimum seizure value is checked.

For example, if 5 seizures have been made to the session agent and none of them have been answered, the sixth time, the session agent is marked as having exceeded its constraints and the calls will not be routed to it until the time-to-resume has elapsed.

- minimum ASR: considered when make routing decisions. If some or all of the calls to the session agent have been answered, the minimum ASR value is considered to make the routing decisions.

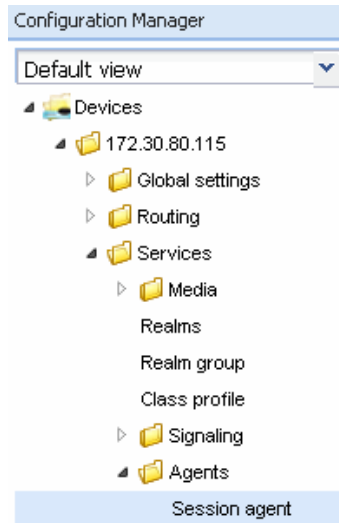
For example, if the you set the minimum ASR at 50% and the session agent's ASR for the current window falls below 50%, the session agent is marked as having exceeded its constraints and calls will not be routed to it until the time-to-resume has elapsed.

Configuring ASR Constraints

You can configure ASR constraints using Net-Net Central.

To configure ASR constraints for SIP and H.323 session agents:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the SIP session agent you want to configure and click **Edit**. The Edit Session agent window appears.
4. Click the SIP session agent constraints tab.
5. **Minimum answer to seizure ratio percentage**—Enter the percentage you want as the minimum. If the session agent's ASR for the current window falls below this percentage, the session agent is marked as having exceeded its constraints and calls will not be routed to it until the time-to-resume has elapsed.
6. **Minimum number of seizures**—Enter the minimum number of seizures that when exceeded, cause the session agent to be marked as having exceeded its constraints. Calls will not be routed to the session agent until the time-to-resume has elapsed. The range of values is 1 to 999999999. The default value is 5.

Answer seizure ratio

Minimum number of seizures	5
Minimum answer to seizure ratio percentage	0

7. Click **Apply**.

SIP Session Agent Continuous Ping

You can configure the Net-Net SBC to use either a keep-alive or continuous method for pinging SIP session agents to determine their health—i.e., whether or not the Net-Net SBC should route requests to them. To summarize the two methods:

- **keep-alive**—The Net-Net SBC sends a ping message of a type you configure to the session agent in the absence of regular traffic.
- **continuous**—The Net-Net SBC sends a ping message regardless of traffic state (regular or irregular); the Net-Net SBC regularly sends a ping sent based on the configured ping interval timer.

How It Works

By sending ping messages, the Net-Net SBC monitors session agents' health and can determine whether or not to take a session out of service (OOS), leave it in service, or bring it back into service after being OOS.

When you set it to use the keep-alive mode of pinging, the Net-Net SBC starts sending a configured ping message to a session agent when traffic for that session agent has become irregular. The Net-Net SBC only sends the ping if there are no SIP transactions with a session agent over a configurable period of time, to which the session agent's response can have one of the following results:

- **Successful response**—A successful response is either any SIP response code or any response code not found in the EMS label: Response codes that put SA in service; these leave the session agent in service. In addition, any successful response or any response in the ping in service response codes parameter can bring a session agent from OOS to in-service status.
- **Unsuccessful response**—An unsuccessful response is any SIP response code configured in the EMS label: Response codes that put SA out of service, and takes the session agent sending it OOS. Because this parameter is blank by default, the Net-Net SBC considers any SIP response code successful.
- **Transaction timeout**—A transaction timeout happens when the session agent fails to send a response to the Net-Net SBC's request, resulting in the session agent's being taken OOS.

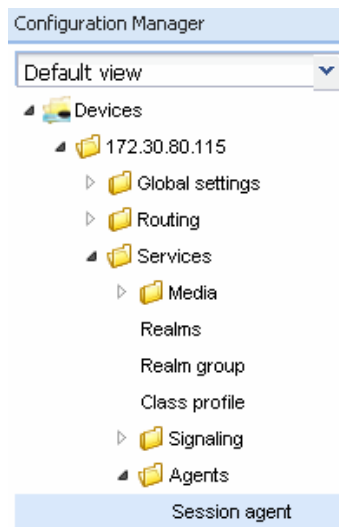
Despite the fact that the keep-alive ping mode is a powerful tool for monitoring session agents' health, you might want to use the continuous ping method if you are concerned about the Net-Net SBC not distinguishing between unsuccessful responses from next-hop session agents and ones from devices downstream from the next-hop session agent. For example, if a SIP hop beyond the session agent responds with a 503 Service Unavailable, the Net-Net SBC does not detect whether a session agent or the device beyond it generated the response.

When you use the continuous ping method, only the next-hop session agent responds—preventing the request from being sent to downstream devices. The Net-Net SBC also sends the ping in regular traffic conditions when in continuous ping mode, so it is certain the response comes from the next hop associated with the session agent. And in continuous ping mode, only entries for the ping out service response codes parameter and transaction timeouts bring session agents OOS.

Configuring the Ping Mode for a Session Agent

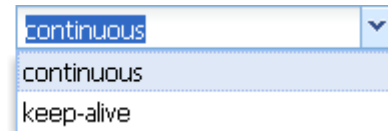
To configure the ping mode for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the SIP session agent routing and address translation tab.
4. Scroll to Test SA availability via SIP pings.
5. **Ping send method**—Select keep-alive or continuous from the combination box. If you want to use continuous ping mode to send ping messages to session agents independently of the background SIP traffic sent to them, set this parameter to continuous. If you want to use the keep-alive mode, leave this parameter set to keep-alive (default).

Ping send method



6. Click Apply.

Session Agent Egress Realm

With session agent egress realm configured, the Net-Net SBC adds a default egress realm to the session agent to identify the signaling interface used for ping requests. The Net-Net SBC also uses the default egress realm when the normal routing request does not yield an egress realm—for example, when a local policy does not specify the next hop's realm.

When you configure session agents, you can define them without realms or you can wildcard the realm value. These are global session agents, and multiple signaling interfaces can reach them. Then, when you use session agent ping, the Net-Net SBC sends out ping requests using the signaling interface of the default egress realm defined in the global SIP configuration. The global session agents in certain environments can cause problems when multiple global session agents residing in multiple networks, some of which might not be reachable using the default SIP interface egress realm.

How It Works

The Net-Net SBC uses the session agent egress realm for ping messages even when the session agent has a realm defined. For normal request routing, the Net-Net SBC uses the egress realm for global session agents when local policies or SIP-NAT bridge configurations do not point to an egress realm.

Ping Message Egress Realm Precedence

For ping messages, the egress realm precedence occurs in the following way (in order of precedence):

- Egress realm identified for the session agent
- Session agent realm (set in the realm-id parameter) or the wildcarded value
- Global SIP configuration egress realm, when configured in the egress-realm parameter
- Global SIP configuration home realm

Normal Request Egress Realm Precedence

For normal request routing, the egress realm precedence occurs in the following way (in order of precedence):

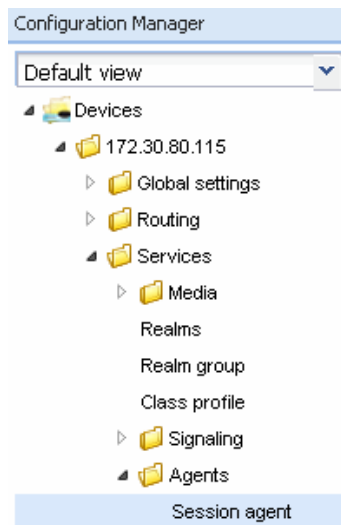
- Egress SIP-NAT realm, when the route home proxy parameter is set to forced and no local policy match is found
- Matching local policy realm, when configured in the local policy attributes
- Session agent realm (set in the realm-id parameter) or the wildcarded value
- Session agent egress realm, when configured in the egress-realm-id parameter
- Global SIP configuration egress realm, when configured in the egress-realm parameter
- Global SIP configuration home realm

Configuring a Session Agent Egress Realm

Configuring a session agent egress realm is optional.

To configure a session agent egress realm:

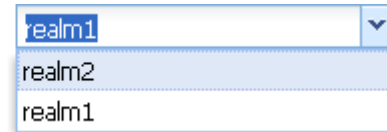
1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.

3. Click the SIP session agent routing and address translation tab.
4. Scroll to Default egress realm for SA pings.
5. **Default egress realm for SA pings**—Click the name of the realm you want defined as the default egress realm used for ping messages from the combination box. The Net-Net SBC will also use this realm when it cannot determine the egress realm from normal routing. There is no default value for this parameter.

Default egress realm for SA pings



6. Click **Apply**.

ENUM Lookup

Telephone Number Mapping (ENUM from TELEphone NUmber Mapping) is a suite of protocols used to unify the telephone system with the Internet by using E.164 addresses with the Domain Name System (DNS). With ENUM, an E.164 number can be expressed as a Fully Qualified Domain Name (FQDN) in a specific Internet infrastructure domain defined for this purpose (e164.arpa). E.164 numbers are globally unique, language independent identifiers for resources on Public Switched Telecommunication Networks (PSTNs). ITU-T recommendation E.164 is the international public telecommunication telephony numbering plan.

How ENUM Works

ENUM uses DNS-based architecture and protocols for mapping a complete international telephone number (for example, +1 202 123 1234) to a series of Uniform Resource Identifiers (URIs).

The protocol itself is defined in the document *E.164 number and DNS* (RFC 3761) that provides facilities to resolve E.164 telephone numbers into other resources or services on the Internet. The syntax of Uniform Resource Identifiers (URIs) is defined in RFC 2396. ENUM uses Naming Authority Pointers (NAPTR) records defined in RFC 2915 in order to identify available ways or services for contacting a specific node identified through the E.164 number.

Translating the Telephone Number

A telephone number is translated into an Internet address using the following steps:

1. The number is first stored in the following format, +1-202-555-1234. 1 is the country code for the United States, Canada, and the seventeen other countries that make up the North American Numbering Plan (NANP). The + indicates that the number is a complete, international E.164 telephone number.
2. All characters are removed except for the digits. For example, 12025551234.
3. The order of the digits is reversed. For example, 43215552021. The telephone number is reversed because DNS reads addresses from right to left, from the most significant to the least significant character. Dots are placed between each digit. Example: 4.3.2.1.5.5.5.2.0.2.1. In DNS terms, each digit becomes a zone. Authority can be delegated to any point within the number.
4. A domain (for example, e164.arpa) is appended to the end of the numbers in order to create a FQDN. For example, 4.3.2.1.5.5.5.2.0.2.1.e164.arpa.

5. The domain name is queried for the resource records that define URIs necessary to access SIP-based VoIP.

Once the authoritative name server for that domain name is found, ENUM retrieves relevant records and uses that data to complete the call or service. For example, the number 12025551234 returns sip:my.name@bigcompany.com.

About NAPTR Records

ENUM uses NAPTR records for URI resource records. NAPTR records are used to translate E.164 addresses to SIP addresses. An example of a NAPTR record is:

```
$ORIGIN 4.3.2.1.5.5.5.2.0.2.1.e164.arpa.
```

```
IN NAPTR 100 10 "u" "sip+E2U" "!\^.*$!sip:phoneme@example.net!"
```

This example specifies that if you want to use the "sip+E2U" service, you should use sip:phoneme@example.net as the address.

The regular expression can be used by a telephone company to easily assign addresses to all of its clients. For example, if your number is +15554242, your SIP address is sip:4242@555telco.example.net; if your number is +15551234, your SIP address is sip:1234@555telco.example.net.

About the Net-Net SBC ENUM Functionality

The ENUM functionality lets the Net-Net SBC make an ENUM query for a SIP request. The ENUM lookup capability lets the Net-Net SBC transform E.164 numbers to URIs during the process of routing (or redirecting) a call. During the routing of a SIP call, the Net-Net SBC uses a local policy attribute to determine if an ENUM query is required and if so which ENUM server(s) need to be queried. A successful ENUM query results in a URI that is used to continue routing or redirecting the call.

ENUM Failover and Query Distribution

The Net-Net SBC can intelligently distribute ENUM queries among all configured ENUM servers. By setting the **Query method** parameter to round-robin, the Net-Net SBC will cycle ENUM queries, sequentially, among all configured ENUM servers. For example, query 1 will be directed to server 1, query 2 will be directed to server 2, query 3 will be directed to server 3, and so on.

The default query method, hunt, directs all ENUM queries toward the first configured ENUM server. If the first server is unreachable, the Net-Net SBC directs all ENUM queries toward the next configured ENUM server, and so on.

Failover to New ENUM Configuration

When an ENUM's configured servers are unreachable via the network, i.e., no response is received on a query, the Net-Net SBC can failover to a defined ENUM configuration that contains different ENUM servers to query. This failover behavior works when all servers in an ENUM config are unreachable, rather than when the Net-Net SBC receives not-found type responses.

The Net-Net SBC queries each ENUM server once before trying the next configured server, and then ultimately trying the servers listed in the **failover-to** ENUM config. If the failover-to servers also are unreachable, the Net-Net SBC fails the call; the failover-to behavior does not recurse among ENUM configs, it only checks the first, linked ENUM config.

ENUM Server Operation States

After five consecutive failed attempts, an ENUM server is considered Out of Service (OOS). All subsequent queries which would be directed to the OOS servers are immediately directed to the first non-OOS server. ENUM servers return to in-service after 600 seconds. If all configured ENUM servers are OOS, the Net-Net SBC fails the call.

After the first failed attempt to reach an ENUM server, it is placed in a Time Out state, which it stays in for 30 seconds. Within this 30 seconds it will not be contacted when an ENUM query is made. After the 30 seconds pass, the ENUM server goes back to an in-service state.

Server Availability Monitoring

The Net-Net SBC can probe an ENUM server's health by sending it a standard ENUM NAPTR query and receiving a valid answer. The query is for the phone number defined in the **health query number** parameter, which should be one that the ENUM servers can positively resolve. As long as the query succeeds, that ENUM server maintains its in-service state and is available for ENUM queries. Any lack of response, whether network based (time-outs), or application based (DNS error or "not found" response) is considered a query failure and the server is set to OOS and unavailable for ENUM queries.

The Net-Net SBC continuously checks the health of all configured ENUM servers to determine their current state and monitor for failed servers' return to service. All servers are checked for availability at the **health query interval** parameter, as defined in seconds.

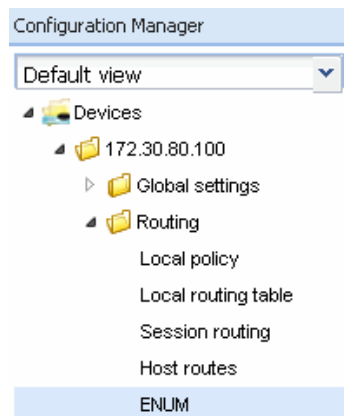
ENUM Server IP Address and Port

You can configure an IP address and port for each enum server listed in the enum-servers parameter. IP address and port are specified in XXX.XXX.XXX.XXX:YYYY format with a port value range of 1024-65535. If the port number is not specified, 53 is assumed.

Configuring ENUM Failover and Query Distribution

To configure ENUM failover and query distribution:

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.
4. Select an ENUM configuration and click **Edit**.

5. **ENUM server**—Click **Add**. The Add instance dialog box appears.
6. **IP address**—Enter the IP address and port value of the ENUM servers (an ENUM server and corresponding redundant servers) to be queried.

The image shows a dialog box titled "Add instance" with a close button (X) in the top right corner. Below the title bar, the text "EnumConfig" is displayed. Underneath, the text "ENUMCONFIG/ENUMSERVERS" is shown. There is a label "*Address" followed by a text input field containing "172.30.251.10". To the right of the input field, the text "(Unique required)" is displayed. At the bottom of the dialog, there are two buttons: "Apply" and "Cancel".

7. Click **Apply**. The IP address is added to the ENUM servers list.
8. Repeat steps 5 through 7 to add additional ENUM servers.
9. Scroll to E164.
10. **Query method**—Select a strategy the Net-Net SBC uses to contact ENUM servers from the combination box. The valid values are:
 - **hunt**—Directs all ENUM queries toward the first configured ENUM server. If the first server is unreachable, the Net-Net SBC directs all ENUM queries toward the next configured ENUM server, and so on.
 - **round-robin**—Cycles all ENUM queries, sequentially, among all configured ENUM servers. Query 1 will be directed to server 1, query 2 will be directed to server 2, query 3 will be directed to server 3.
11. **Health query number**—Enter a standard ENUM NAPTR query that will consistently return a positive response from the ENUM server.
12. **Health query interval (sec)**—Enter the number of seconds to perpetually probe ENUM servers for health. The valid values are **0** through **65535** seconds. The default value is **0**.
13. **Failover ENUM configuration**—Click the name of another ENUM configuration which to failover to under appropriate conditions in the combination box.

E164

The image shows a configuration form for E164. It has four rows, each with a label on the left and a control on the right. The first row is "Query method" with a dropdown menu showing "hunt". The second row is "Health query number" with a text input field containing "12345". The third row is "Health query interval(sec)" with a text input field containing "500". The fourth row is "Failover ENUM configuration" with a dropdown menu showing "ENUMtest" (highlighted) and "ENUMtest" below it.

14. Click **Apply**.

Source URI Information in ENUM Requests

ENUM queries can include the source URI that caused the ENUM request by enabling the **include source info** parameter. The Net-Net SBC can add the P-Asserted-ID URI (only if not in an INVITE) or the From URI into an OPT-RR Additional Record to be sent to the ENUM server. It can be useful to specify the originating SIP or TEL URI from a SIP request which triggered the ENUM query, so the ENUM server can provide a customized response based on the caller.

This feature implements the functionality described in the Internet Draft, *DNS Extension for ENUM Source-URI*, draft-kaplan-enum-source-uri-00.

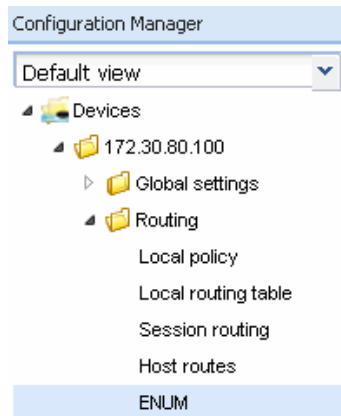
When a P-Asserted-ID is blocked or removed before the ENUM query is made, the Net-Net SBC only sends the URI in the From header.

Note that to support this feature, according to the Internet draft, ENUM clients must support 1220 bytes in UDP responses. Therefore, if this feature is enabled, and the max response size parameter is not set i.e., with a 512 byte default, the Net-Net SBC will set the size to 1200 on the OPT-RR records sent.

Configuring OPT-RR Additional Records Sent to ENUM Servers

To send OPT-RR additional records to ENUM servers:

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.

4. **Include source information in queries**—Select enabled from the combination box to enable the Net-Net SBC to send source URI information to the ENUM server with any ENUM queries.

E164

Query method	hunt
Health query number	12345
Health query interval(sec)	500
Failover ENUM configuration	ENUMtest ENUMtest

5. Click **Apply**.

Caching ENUM Responses

As DNS responses often lead to further DNS queries, a DNS server can send additional multiple records in a response to attempt to anticipate the need for additional queries. The Net-Net SBC can locally cache additional NAPTR, SRV, and A records returned from an ENUM query to eliminate the need for unnecessary external DNS requests by enabling the **Cache additional DNS records** parameter. These cached records can then be accessed by internal ENUM and DNS agents.

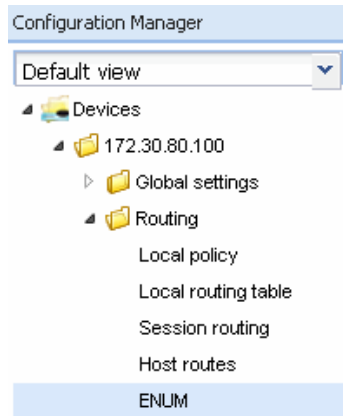
The unprompted NAPTR, SRV, or A record returned to the Net-Net SBC must include complete information to resolve a call to be added to the local DNS/ENUM cache, otherwise the Net-Net SBC will perform an external query to find the address it is looking to resolve.

Cached entries are per ENUM configuration. That means if one ENUM configuration has a number of cached entries, and an ENUM request is directed through a different ENUM configuration, the second configuration is not privy to what the first configuration has cached.

The Net-Net SBC uses the shorter lifetime of the DNS response's TTL or the server dns attribute's transaction-timeout to determine when to purge a DNS record from the local cache.

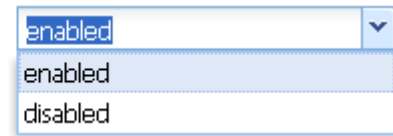
Adding Additional Records to the Local Cache

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.
4. **Cache additional DNS records**—Select enabled from the combination box to enable the Net-Net SBC to add additional records received in an ENUM query to the local DNS cache.

Include source information in queries



5. Click **Apply**.

Operation Modes

There are four modes of ENUM operation that are selected on a global basis:

- stateless proxy
- transaction stateful proxy
- session stateful proxy
- B2BUA with or without media

Stateless Proxy Mode

The stateless proxy mode is the most basic form of SIP operation. The stateless proxy mode:

- Has the least number of messages per call. No record route header is added and there are no 100 Trying or BYEs.
- Does not keep transaction state (timers and retransmission). There are no session counters and no session stop time. No session stop time means no RADIUS STOP records.
- Has no limits on session state.
- Can restrict functionality by specification. This can mean no media management, limited potential for RADIUS accounting, and no CALEA (no Release/BYE messages for CDC).
- Acts primarily as a routing device, with local policy routing and ENUM routing.

Transaction Stateful Proxy

In the transaction stateful proxy mode:

- Adds state to the proxy (not dialogs).
- Has lower number of messages per call. No Record Route header added and no BYES.
- Keeps transaction state (timers and retransmissions).
- Enforces session restrictions (32k) because of state management. These restrictions can be increased.
- Can restrict functionality by specification. This can mean no media management, limited potential for RADIUS accounting, and no CALEA (no Release/BYE message for CDC).
- Acts as routing device with transaction timers, with local policy routing and ENUM routing.
- Can off-load some transactions across unreliable links.

Session Stateful Proxy

The session stateful proxy mode:

- Maintains dialog state as a proxy.
- Includes BYES (though cannot be inserted)
- Keeps transaction state (timers and retransmission)
- Provides per-session information such as session counters per session agent, RADIUS STOP accounting record generation, CALEA CDC generation.
- Enforces session restrictions (32k) because of state management.
- Does not provide media management. There is no CALEA CCC.
- Routes full sessions with transaction timers with local policy routing and ENUM routing.

B2BUA

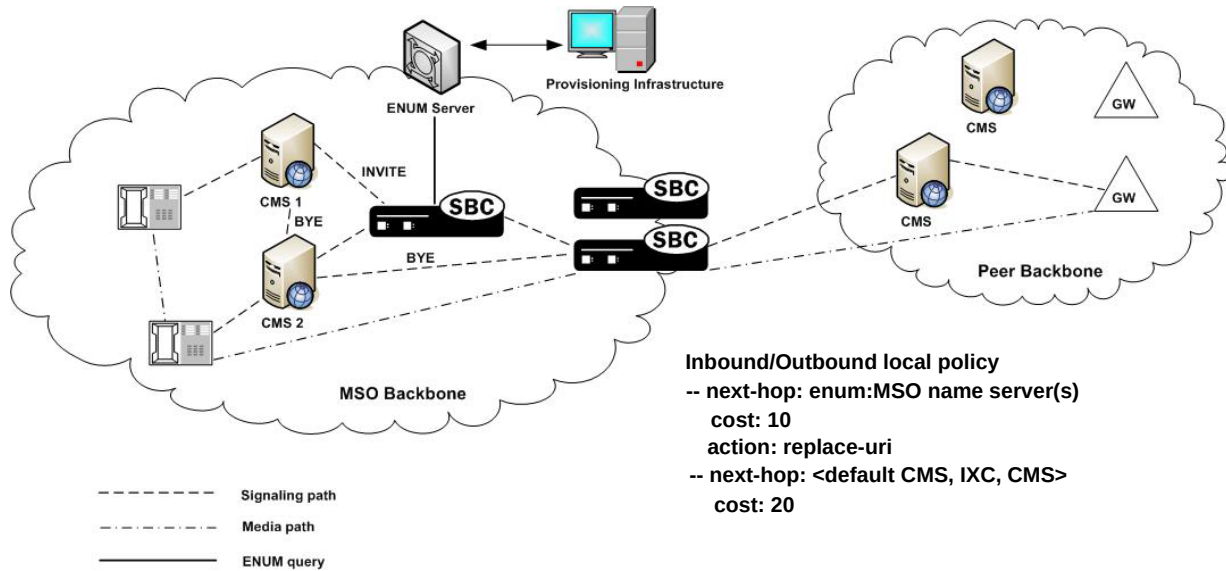
The B2BUA mode:

- Acts as UAS and UAC within call flow.
- Includes BYES (can be inserted).
- Keeps transaction state (timers and retransmissions)
- Provides per-session information such as session counters per session agent, RADIUS STOP accounting record generation, CALEA CDC generation.
- Enforces session restrictions (32k) because of state management.
- Can provide media management, including media routing through a single IP address with topology masking, CALEA CCC, media watchdogs for state management.
- Routes full sessions with topology masking. Includes rewriting Via, Route, Contact headers, full NATing with SIP NAT or header manipulation, direct bridging, local policy routing, and ENUM routing.

Example: ENUM Stateless Proxy

The following diagram shows the Net-Net SBC using ENUM to query a local subscriber database. The Net-Net SBC serves as the inbound and outbound routing

hub and performs media management. Calls are routed throughout the MSO network using ENUM lookup results.

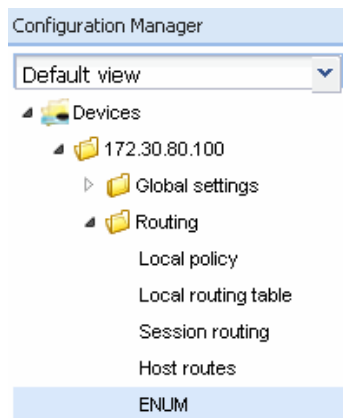


Configuring ENUM

You can use the Net-Net Central to configure ENUM.

To configure ENUM:

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.
4. Click **Add**. The Add ENUM config dialog box appears.
5. **ENUM config name**—Enter a string that uniquely identifies this ENUM configuration. You use this name in other areas of the Net-Net SBC configuration to refer to this ENUM configuration.
6. Select the ENUM configuration from the table and click **Edit**.
7. **Top level domain**—Enter the domain extension to be used when querying the ENUM servers for this configuration. For example, e164.arpa. The query name is a concatenation of the number and the domain.

8. **Realm name**—Choose the realm where the ENUM servers can be reached from the combination box. The realm ID is used to determine on which network interface to issue the ENUM query.

*ENUM configuration name	ENUMtest3
Top level domain	e164.arpa
Realm name	realm1 ▼

9. **ENUM query lookup length (digits)**—Specify the length of the ENUM query, starting from the most significant digit. This provides for more efficient caching of URI lookup results. The valid range for this value is 1-255.

ENUM query lookup length (digits)	10

10. **Query timeout period (sec)**—Enter the total time in seconds that should elapse before a query sent to a server (and its retransmissions) will timeout. If the first query times out, the next server is queried and the same timeout is applied. This process continues until all the servers in the list have timed out or until one of the servers responds.

The retransmission of ENUM queries is controlled by three timers. These timers are derived from this timeout value and from underlying logic that the minimum allowed retransmission interval should be 250 milliseconds; and that the Net-Net SBC should retransmit 3 times before timing out to give the server a chance to respond.

- Init-timer is the initial retransmission interval. If a response to a query is not received within this interval, the query is retransmitted. To safeguard from performance degradation, the minimum value allowed for this timer is 250 milliseconds.
- Max-timer is the maximum retransmission interval. The interval is doubled after every retransmission. If the resulting retransmission interval is greater than the value of max-timer, it is set to the max-timer value.
- Expire-timer: is the query expiration timer. If a response is not received for a query and its retransmissions within this interval, the server will be considered non-responsive and the next server in the list will be tried.

The following examples show different timeout values and the corresponding timers derived from them.

timeout >= 3 seconds

```
Init-timer = Timeout/11
Max-Timer = 4 * Init-timer
Expire-Timer = Timeout
```

timeout = 1 second

```
Init-Timer = 250 ms
Max-Timer = 250 ms
Expire-Timer = 1 sec
```

timeout = 2 seconds

```
Init-Timer = 250 ms
Max-Timer = 650 ms
Expire-Timer = 2sec
```

11. **Cache inactivity timer (sec)**—Enter the time interval in seconds after which you want cache entries created by ENUM requests deleted, if inactive for this interval. If the cache entry gets a hit, the timer restarts and the algorithm is continued until the cache entry reaches its actual time to live.

Setting this value to zero disables caching. Rarely used cache entries are purged and frequently used entries are retained.

Query timeout period (sec)	11
Cache inactivity timer (sec)	3600

12. Click **Apply**. The newly created ENUM configuration appears in the Enum config table.

Custom ENUM Service Type Support

You can configure the ENUM service type that you want to use for an ENUM group. The Net-Net SBC has always supported **E2U+sip** and **sip+E2U** by default, and still does. However, you are also able to configure the service type to those supported in RFCs 2916 and 3721.

For example, you can now set the service type in the ENUM configuration to support **E2U+sip** and **E2U+voicemsg:sip**. When you configure customer ENUM service types on your system, however, you should note the following:

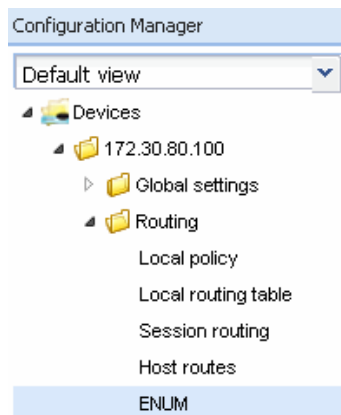
- New entries in the **service type** parameter overwrite pre-existing values, including the default values.
- Because of the overwriting noted above, you must include the defaults (if you want them configured) when you are adding additional ENUM service type support. That is, you have to also type in **E2U+sip** and **sip+E2U** if you want them to be used in addition to the customized types you are setting.

Configuring Custom ENUM Support

You configure custom ENUM service type support in the ENUM configuration. The **service type** parameter takes more than one value. The following example shows you how to enter multiple service types.

To support custom ENUM service types:

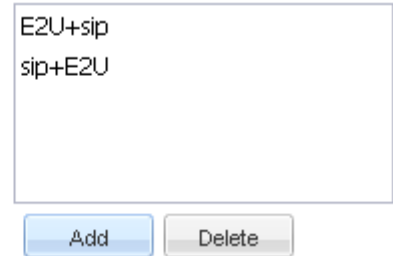
1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.

3. Select an ENUM configuration and click **Edit**.
4. **Service type**—Click **Add**. The Service type window appears.
Enter the ENUM service types you want supported in this ENUM configuration.
Possible entries are **E2U+sip** and **sip+E2U** (the default), and the types outlines in RFCs 2916 and 3721. This parameter defaults to the following service types:
E2U+sip and **sip+E2U**.

Service type



The screenshot shows a rectangular window titled 'Service type'. Inside the window, there is a list containing two entries: 'E2U+sip' and 'sip+E2U'. Below the list, there are two buttons: a blue 'Add' button and a grey 'Delete' button.

5. Click **OK**.
6. Repeat steps 5 and 6 to enter multiple service types.
7. Click **Apply**.

Editing ENUM Service Types

You can edit the ENUM service type.

To edit an ENUM service type:

1. In the ENUM configuration table, click the ENUM configuration you want to edit the **service type** for and click **Edit**. The Edit ENUM configuration window appears.
2. **Service type**—Select the service type you want to edit and click **Edit**. The Service type window appears.
3. Edit your service type.
4. Click **OK**.
5. Click **Apply**.

Deleting ENUM Service Types

You can delete the ENUM service type.

To delete an ENUM service type:

1. In the ENUM configuration table, click the ENUM configuration whose **service type** you want to delete and click **Edit**. The Edit ENUM configuration window appears.
2. **Service type**—Select the service type you want to delete and click **Delete**. The service type is deleted from the service type list.
3. Click **Apply**.

Configuring Local Policy Attributes

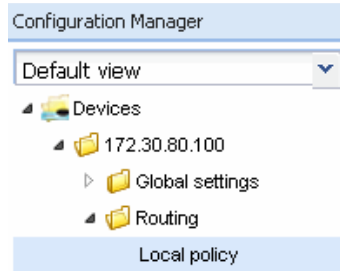
You can specify that an ENUM query needs to be done for the routing of SIP calls. You do so by configuring the local policy's next-hop attribute with the name of a specific ENUM configuration, prefixed with the **enum:** tag. For example: **enum:test**.

You can configure multiple next-hops with different ENUM servers or server groups (possibly with different top-level-domains). If the first ENUM server group you enter as the next hop is not available, one of the others can be used.

Note: A new parameter called action has replaced the policy attribute's replace-uri parameter available prior to build 211p19.

To configure local policy attributes:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Click **Add** in the Routes window. The Add route dialog box appears.
4. **To Address**—Enter the To IP address.
5. **From Address**—Enter the From IP address.
6. **Source realm**—Type the Source realm in the field.
7. Click **Apply**.

 A screenshot of the 'Add Local policy' dialog box. It has a title bar with 'Add Local policy' and a close button. The dialog contains three sections:
 1. '(Number/address/host)' section with a label '*To address', an input field, and a 'Configuration Guidelines' dropdown.
 2. '(Number/address/host)' section with a label '*From address', an input field, and a 'Configuration Guidelines' dropdown.
 3. 'Source realms' section with a label '*Source realm' and an input field.
 At the bottom are 'Apply' and 'Cancel' buttons.

8. Select the Local policy configuration from the table and click **Edit**.
9. Scroll to Route policy attributes and click **Add**.
10. **Next hop**—Enter the ENUM configuration as the next hop.
11. **Action**—Select redirect from the combination box to redirect (if you want to send a REDIRECT message back to the calling party with the information

returned by ENUM in the Contact). The calling party then needs to send an INVITE using that information.

Add Route policy attributes

*Start time: 0000

*End time: 2400

*Days of the week: U-S

*Next hop: SAG:SAGGroup

*Action: **redirect** (dropdown menu open showing: none, redirect, replace-uri, SIP)

*Terminate recursion:

*Egress protocol:

Apply Cancel

12. Click **Apply**.
13. Click **Apply** in the content area.

CNAM Subtype Support for ENUM Queries

Calling Name (CNAM) data is a string of up to 15 ASCII characters of information associated with a specific calling party name. The Internet-draft, draft-ietf-enum-cnam-08.txt, registers the Enumservice 'pstndata' and subtype 'cnam' using the URI scheme 'pstndata:' to specify the return of CNAM data in ENUM responses. The Net-Net SBC recognizes CNAM data returned via this mechanism. CNAM data is then inserted into the display name of the From: header in the original Request. If a P-Asserted-ID header is present in the original request, the CNAM data is inserted there as well.

CNAM data is identified by an ENUM response with service type:

E2U+pstndata:cnam

CNAM support is configured in the Profiles configuration and then is applied to either a session agent, realm, or SIP interface.

The Net-Net SBC can perform CNAM queries on the signaling message's ingress or egress from the system by setting the **Lookup direction** parameter to either ingress or egress. If the CNAM **Lookup direction** parameters are configured on both the ingress and egress sides of a call, the Net-Net SBC will only preform the lookup on the ingress side of the call.

CNAM Unavailable Response

A CNAM response can include a Calling Name Privacy Indicator parameter ('unavailable=p') or Calling Name Status Indicator parameter ('unavailable=u') in responses. The Net-Net SBC can insert a custom reason string into the SIP message's From and P-Asserted-ID header in the original requires.

Configuring the **Unavailable ptype reason** parameter inserts the specified text into the From and P-Asserted-ID headers when a CNAM response contains the unavailable=p parameter.

Configuring the **Unavailable utype reason** parameter inserts the specified text into the From and P-Asserted-ID headers when a CNAM response contains the unavailable=u parameter.

SIP Profile Inheritance

CNAM features, via the SIP Profile configuration, can be applied to session agents, realms, and SIP interfaces. The more generalized object inherits the more specific object's values. For example, if CNAM support via a SIP profile is configured on a session agent, the expected processing will override any SIP profile configuration on the downstream realm or SIP interface. Likewise, if CNAM support is not configured on the receiving session agent, but configured in the realm, CNAM configuration on the SIP interface will be ignored.

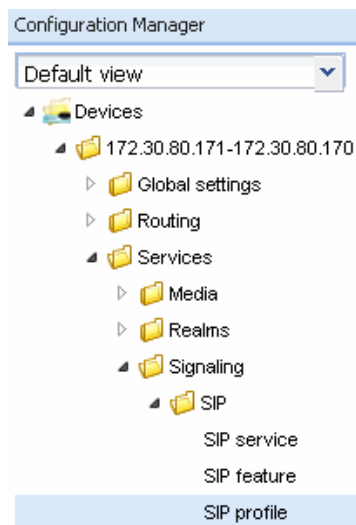
Configuring CNAM Subtype Support

To enable the Net-Net SBC to preform CNAM subtype ENUM queries, you must configure a SIP profile with an ENUM configuration that points to valid ENUM servers. The referenced ENUM config configuration lists the servers to contact for CNAM type queries (and other general ENUM server interaction parameters).

To configure CNAM subtype support:

To create a SIP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.
3. Click **Add**. The Add SIP profile dialog box appears.

4. **Name**—Enter the name of the SIP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.

The dialog box is titled "Add instance" and contains a section labeled "SIPPROFILE". Inside this section, there is a label "*Name" followed by a text input field containing "sipProfile1". To the right of the input field is the text "(Unique required)". At the bottom of the dialog box are two buttons: "Apply" and "Cancel".

5. Click **Apply**. The new SIP profile appears in the SIP profile table.
6. Click the SIP profile you want to configure and click **Edit**. The Edit SIP profile dialog box appears.
7. Scroll to CNAM.
8. **Lookup ENUM configuration**—Select the name of an ENUM configuration from the combination box that will query ENUM servers for CNAM data.
9. **Lookup direction**—Select **ingress** or **egress** from the combination box to identify where the Net-Net SBC performs a CNAM lookup with respect to where the call traverses the system. The default value is **egress**.
10. **Unavailable ptype reason**—Enter a string, no more than 15 characters, to indicate that the unavailable=p parameter was returned in a CNAM response.
11. **Unavailable utype reason**—Enter a string, no more than 15 characters, to indicate that the unavailable=u parameter was returned in a CNAM response.

CNAM

Lookup ENUM configuration	ENUM1
Lookup direction	egress
Unavailable ptype reason	unavailable=p reason
Unavailable utype reason	unavailable=u reason

12. Click **Apply**.

ENUM/Local Route Table Support for H.323 and IWF

Local Route Table (LRT) support for H.323 and IWF is compatible with that currently offered for SIP. LRT and ENUM provide the Net-Net SBC with the ability to perform routing based on ENUM queries to a DNS server or local to an onboard database.

For the LRT feature, this means that entries in the local routing table now include those prefixed with the **h323:** URI scheme, indicating that H.323 is the next hop protocol.

IWF Considerations

When the system performs a local policy lookup for an incoming SIP or H.323 call and determines an ENUM/LRT server is the next hop, it queries that ENUM/LRT server. The response will include the URI scheme, indicating the next hop protocol and the hostname/IP address representing the next hop. For cases where the

incoming call signaling protocol and the URI scheme of the ENUM/LRT response are the same, the call requires no interworking. The Net-Net SBC can simply route the egress call leg to the specified next hop.

However, interworking is required when the incoming signaling protocol and the URI scheme of the ENUM/LRT response do not match. In these cases, the Net-Net SBC interworks between SIP and H.323, routing the call to the appropriate next hop.

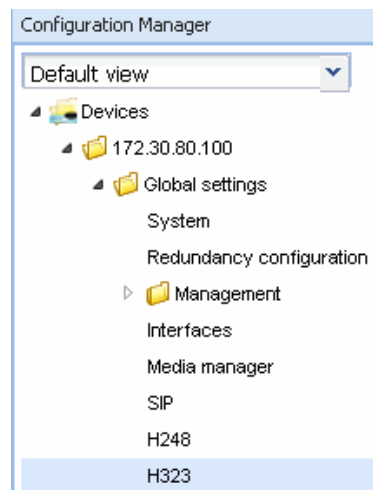
In addition, the Net-Net SBC compares the URI scheme returned in the ENUM/LRT response to the application protocol specified in the policy attributes. If the URI scheme is SIP but the policy attributes indicate H.323, the route is deemed invalid. The same is true for an H.323 URI scheme and SIP route.

Configuring LRT Support

In order for LRT to work for H.323 and IWF calls, you do not have to perform any special configuration. However, you can configure the system to match ENUM/LRT responses against session agent groups, and then use those SAGs for routing.

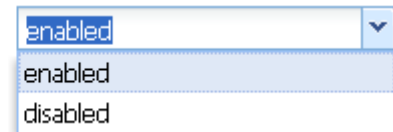
To enable matching ENUM/Local Route Table responses for H.323 SAG routing:

1. Expand the Configuration Manager slider > Global settings.



2. Click H323.
3. Press **Add** to add the H.323 configuration to the content area.
4. **ENUM SAG match**—Select enabled from the combination box to allow the Net-Net SBC to perform matching against the hostnames in ENUM/LRT lookup responses and session agent groups. If there is a match, the Net-Net SBC uses the matching SAG for routing. If no match is found, normal ENUM/LRT routing proceeds.

ENUM SAG match



5. Click **Apply**.

P-CSCF Redundancy with Implicit Service Route

You can configure multiple P-CSCFs in a core network to provide redundant P-CSCF functionality. Configuring multiple P-CSCFs is performed similarly to configuring any type of redundant session agent (SA), either through DNS for a single SA or with a session agent group (SAG). If an SA (or member of an SAG) goes out of service, and the IP address is unreachable, the Net-Net SBC uses the next IP address for the SA (or member of an SAG). Also the Net-Net SBC can load balance P-CSCFs by using a load balancing SAG strategy such as round-robin.

Tracking UE per P-CSCF

To ensure continuity of service, the Net-Net SBC tracks the relationship between a UE and the P-CSCF it registers to. When a P-CSCF goes out of service and fails over to the next P-CSCF (as an SA or member of an SAG), the Net-Net SBC notes the time of the failover event. When a UE associated with the failed P-CSCF re-registers, the Net-Net SBC sees that the previous P-CSCF has experienced a failover. Upon a P-CSCF failover, the Net-Net SBC redirects the associated UE's traffic to the new P-CSCF as the registrar/proxy.

To configure the Net-Net SBC to direct a UE to use a P-CSCF that has been placed in service after the previous P-CSCF failed, use the parameter that enables the implicit service-route feature as described in the previous section.

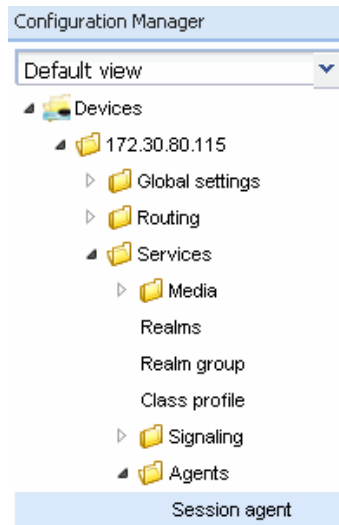
Registration Invalidation

The Net-Net SBC includes provisions to explicitly invalidate current registrations in certain instances. If the P-CSCF (configured as a SA) goes out of service, or if the active IP address that specifies a P-CSCF does not point to the active P-CSCF, the Net-Net SBC can invalidate the current endpoint registrations to that SA. The new configuration, when enabled, forces the Net-Net SBC to act as if the registrations have expired, thus forcing the endpoints to re-register. The Net-Net SBC responds back to the registering endpoint with a 403 response (if the allow-anonymous parameter in the sip-ports configuration element is set to registered).

Configuring P-CSCF Redundancy with Implicit Service Route

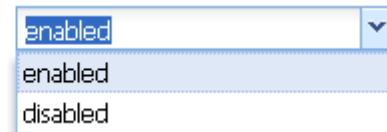
To enable registration invalidation support:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a SIP session agent from the Session agents table and click **Edit**.
4. Scroll to Ping parameters.
5. **Invalidate registrations**—Choose **enabled** from the combination box.

Invalidate registrations



6. Click **Apply**.

Route Recursion Override

The route recursion override features lets you stop route recursion when the Net-Net SBC performs local policy lookups. It prevents the Net-Net SBC from including lower-priority routes in the target set (the set of next hop targets) of next hop destinations for routes that are configured to stop recursion.

Normally, the Net-Net SBC uses the local policy to determine a target set of next hops that match the selection criteria. It orders the routes in the target set from highest to lowest priority based on:

- Cost
- Best match (source realm, From address, and To address)
- Time of day
- Day of week
- Media profiles

Without the route recursion override feature enabled, all routes are included in the target set. However, there are cases when you might not want the lower priority routes included in the target set.

For example, calls to a specific number prefix have a local policy associated with them so that calls with this prefix are routed to session agents that apply constraints preventing overload. If a route exceeds the constraints and has been excluded on that basis, the Net-Net SBC will add another less-specific matching local policy (if one is available) to the target set. This lower-priority route allows the request to be forwarded, and therefore defeats the constraints that prevent overload conditions.

Another example is to create more specific routes to filter and block traffic by setting the next-hop to a fake session agent. For example, you can route all 123* prefix calls, but block 12345* prefix numbers.

How It Works

When you enable route recursion override, all routes of lower priority in the target set are discarded. In essence, you flag a specific route and so prevent recursion to additional lower-priority routes that are less specific.

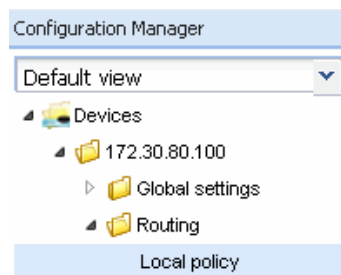
This feature is enabled in the local policy attributes, where you can set a parameter that causes the route represented by the local policy attribute to stop the recursion. If the Net-Net SBC finds an entry where the route recursion override feature is enabled, it does not lookup or use any less specific or lower priority routes.

Configuring Route Recursion Override

You can configure route recursion override using the Net-Net Central.

To enable route recursion override for a route represented by a local policy attribute:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Click a route name in the table to select it and click **Edit**.
4. Scroll to Route policy attributes and click **Add**.
5. **Next hop**—Enter the next signaling host IP address, SAG, hostname, or ENUM config.

6. **Terminate recursion**—Select enabled from the combination box to terminate route recursion with this next hop.

*Next hop	*
*Action	none
*Terminate recursion	enabled
*Egress protocol	disabled

7. Click **Apply**.
8. Click **Apply** in the content area.

Codec Reordering

Certain carriers deploy voice services where their peering partners do not use the carriers' preferred codecs. The Net-Net SBC can now reorder the codecs so that the preferred one is selected first.

Take the example of a carrier that deploys a voice service using G.729 rather than G.711. If that carrier has a peering partner providing call origination for the VoIP customers with G.711 used as the preferred codec, there can be issues with codec selection.

The Net-Net SBC can reorder the default codec in an SDP offer to the preferred codec before it forwards the offer to the target endpoint. When you enable this feature, you increase the probability that the target endpoint will choose the preferred codec for its SDP answer, thereby avoiding use of the undesired codec.

How It Works

You enable the codec reordering feature by setting the `preferred-codec=x` (where X is the preferred codec) option in the realm and session agent configurations. You set it in the realm from which the Net-Net SBC receives SDP offers (in requests or responses), and for which the media format list needs to be reordered by the Net-Net SBC prior to being forwarded. To configure additional codec ordering support for cases when a response or request with an SDP offer is from a session agent, you can set this option in the session agent configuration.

If you enable the option, the Net-Net SBC examines each SDP media description before it forwards an SDP offer. And if necessary, it performs reordering of the media format list to make the preferred codec first in the list of the SDP m= line.

The Net-Net SBC determines preferred codecs in the following ways:

- If the response or request with an SDP offer is from a session agent, the Net-Net SBC determines the preferred codec by referring to the session agent configuration. You set the preferred codec for a session agent by configuring it with the `preferred-codec=x` option.

- If the response or request with an SDP offer is not from a session agent or is from a session agent that does not have the `preferred-codec=x` option configured, the Net-Net SBC determines the preferred codec by referring to the `preferred-codec=x` option in the realm.
- If the Net-Net SBC cannot determine a preferred codec, it does not perform codec reordering.

The way that the Net-Net SBC performs codec reordering is to search for the preferred codec in the SDP offer's media description (`m=`) line, and designate it as the codec of choice (if it is not already so). After it marks the preferred codec as the default, the Net-Net SBC does not perform any operation on the remaining codecs in the media format list. Note that the Net-Net SBC performs codec reordering on the media format list only. If the `rtptime` attribute of the preferred codec is present, the Net-Net SBC does not reorder it.

Preferred Codec Precedence

When you configure preferred codecs in session agents and realms, be aware that the codec you set for a session agent takes precedence over one you set for a realm. This means that if you set preferred codecs in both configurations, the one you set for the session agent will be used.

In the case where the Net-Net SBC does not find the session agent's preferred codec in the SDP offer's media format list, then it does not perform codec reordering even if the media format list contains the realm's preferred codec.

Configuring Codec Reordering

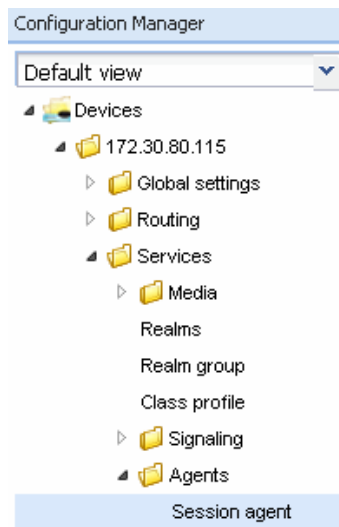
When you configure codec ordering, the codec you set must match the name of a media profile configuration. If your configuration does not use media profiles, then the name of the preferred codec that you set must be one of the following:

- PCMU
- G726-32
- G723
- PCMA
- G722
- G728
- G729

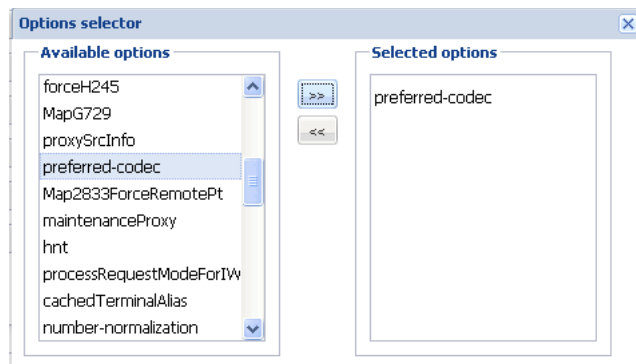
Note: When you configure this feature for a session agent, you must configure it for the associated realm as well. Otherwise, the feature will not work correctly.

To set a preferred codec for a session agent configuration:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a SIP session agent from the Session agents table and click **Edit**.
4. Scroll to the SIP Options for this SA section and click **Options**.
5. Choose the **preferred-codec** option from the Available options list.



6. Click  to move the preferred-codec option to the Selected options list. The option name and description appear in the middle of the screen.
7. Click **OK**.
8. Click **Apply**.

QoS Based Routing

In addition to configuring your system for routing based on certain session constraints, you can also set up routing based on QoS. QoS based routing uses the R-Factor on a per-realm basis to either cut back on the traffic allowed by a specific realm, or to shut that traffic off altogether.

To use this feature, you set up QoS constraints configurations and apply one per realm. The QoS constraints configuration allows you to set up two thresholds:

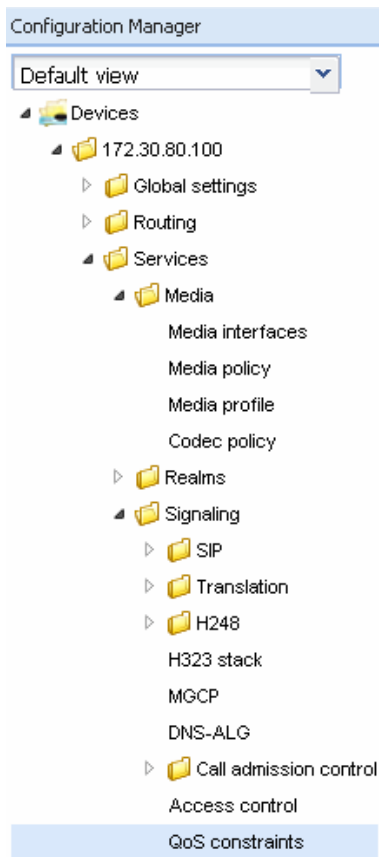
- **Major**—The major threshold sets the R-Factor limit beyond which the Net-Net SBC rejects a certain percentage (that you configure) of calls. That is to say, it rejects inbound calls at the rate you set with a **503 Service unavailable** status code, and rejects outbound calls if there are no alternative routes.
- **Critical**—The critical threshold, when exceeded, causes the Net-Net SBC to behave the same way it does when any of the session constraints (set in the session-constraints configuration) are exceeded. All inbound calls to the realm are rejected with a **503 Service unavailable** status code, and (if there is no alternate route) outbound calls are rejected, too. Until the R-Factor falls within acceptable means and the session constraint's time-to-resume value has elapsed, the realm remains in this state.

Configuring QoS Constraints

Your first step to enable QoS based routing is to set up a QoS constraints configuration. This configuration is where you enter major and critical thresholds, as well as the load reduction for the realm should the R-Factor exceed the major threshold. You then apply it to a realm.

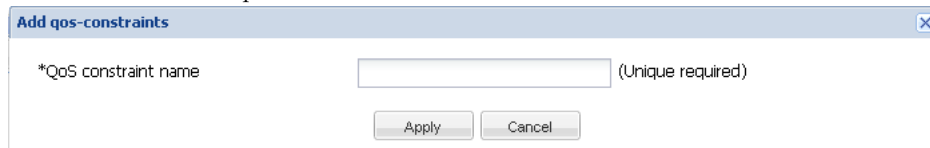
To set up a QoS constraints configuration:

1. Expand the Configuration Manager slider > Services > Signaling.



2. Click QoS Constraints.
3. Click **Add**. The Add QoS Constraint dialog box appears.

4. **Name**—Enter the unique name of this QoS constraints configuration. You use this name when applying the configuration to a realm. This parameter has no default and is required.



A dialog box titled "Add qos-constraints" with a close button (X) in the top right corner. It contains a text input field labeled "*QoS constraint name" with the text "(Unique required)" to its right. Below the input field are two buttons: "Apply" and "Cancel".

The QoS constraint is added to the QoS constraint table.

5. Click the QoS constraint you want to configure and click Edit. The Edit QoS Constraint dialog box appears.
6. **Enable QoS constraint**—Select enabled from the combination box.
7. **Major RFactor threshold**—Enter a numeric value between 0 (default) and 9321 to set the threshold that determines when the Net-Net SBC applies the call reduction rate. If you leave this parameter set to 0, the Net-Net SBC will not apply a major threshold for any realm where you apply this QoS constraints configuration.

This value must be less than what you set for the **Critical RFactor threshold**, except when the **Major RFactor threshold** is 0.

8. **Critical RFactor threshold**—Enter a numeric value between 0 (default) and 9321 to set the threshold that determines when the Net-Net SBC rejects all inbound calls for the realm, and rejects outbound calls when there is no alternate route. If you leave this parameter set to 0, then the Net-Net SBC will not apply a critical threshold for any realm where you apply this QoS constraints configuration.

This value must be greater than what you set for the **Major RFactor threshold**, except when the **Major RFactor threshold** is 0.

9. **Percentage of call load reduction**—Enter a number from 0 (default) to 100 representing the percentage by which the Net-Net SBC will reduce calls to the realm if the **Major RFactor threshold** is exceeded. If you leave this parameter set to 0, then the Net-Net SBC will not reduce call load for the realm—even when the **Major-RFactor** is configured.

This is the percentage of inbound and outbound calls the Net-Net SBC will reject. For example, if you set this parameter to 50 and the major threshold is exceeded, then the Net-Net SBC rejects every other call to the realm.

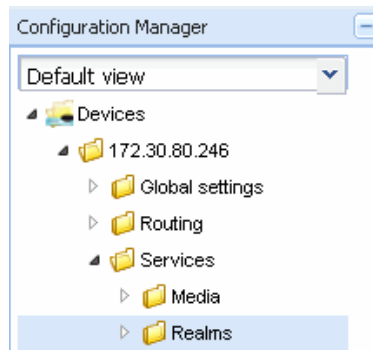
*QoS constraint name	qos1
Enable QoS constraint	enabled ▼
Major Rfactor threshold	0
Critical Rfactor threshold	0
Percentage of call load reduction	100

10. Click **Apply**.

Applying QoS Constraint to a Realm

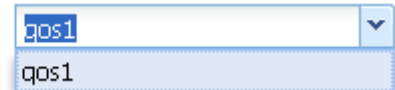
To apply a QoS constraint to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to apply QoS constraints to and click **Edit**.
4. Scroll to QoS constraints.
5. **QoS constraints**—Click the QoS constraint you want to apply to this realm from the combination box.

QoS constraints



6. Click **Apply**.

Multi-Stage Local Policy Routing for SIP

Multistage local policy routing enables the Net-Net SBC to perform multiple stages of route lookups where the result from one stage is used as the lookup key for the next routing stage.

Routing Stages

A routing stage signifies a re-evaluation of local policy based on the results of a local policy lookup. In the simplest, single stage case, the Net-Net SBC performs a local policy lookup on a SIP message's Request URI. The result of that local policy lookup is a next hop FQDN, IP address, ENUM lookup, or LRT lookup; that result is where the Net-Net SBC forwards the message. In the multistage routing model, that resultant next hop is used as the lookup key for a second local policy lookup.

The results of each stage do not erase the results of the previous stage. Thus, previous results are also possible routes to use for recursion, but the next stage results are tried first.

Note: Setting a next hop to a SAG in a multistage scenario constitutes an error.

Network Applications

The following are typical applications of multistage routing:

- An operator might need to query an ENUM server for a destination number. Based on the NAPTR result of the ENUM query, the Net-Net SBC performs a

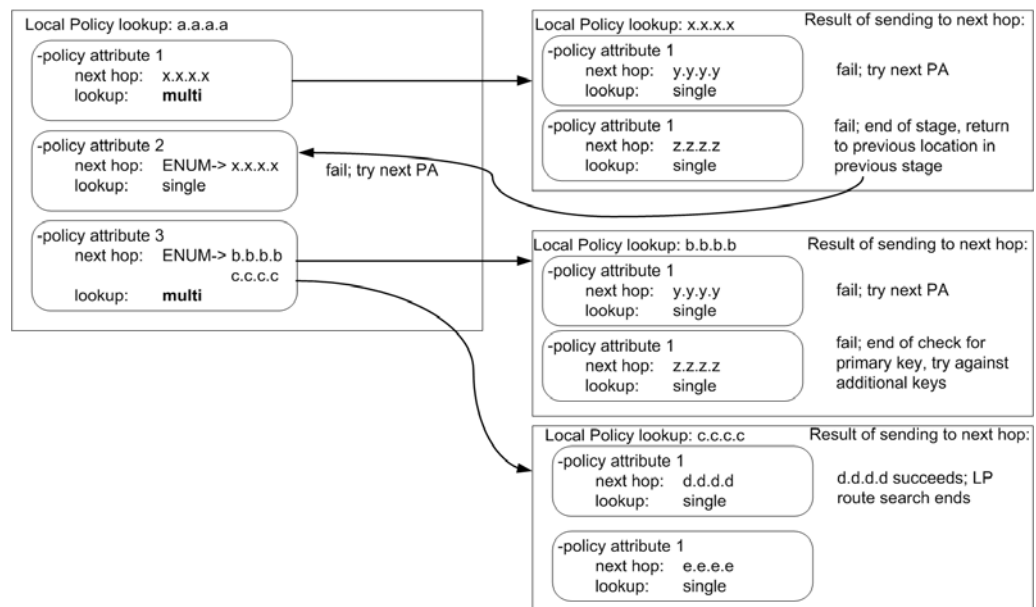
local policy lookup to decide how to route the request, perhaps based on a LRT table lookup.

- An operator might need to query one ENUM server for a number portability lookup, then based on the routing number perform a second ENUM query to a different server to learn which carrier to use for the routing number. Then, then based on the identified carrier perform a LRT lookup for what next-hop(s) to use for that carrier.
- An operator might query an LRT table to confirm the allowed source number. Then, based on the result, query an ENUM server for destination routing.

Multistage Routing Conceptual Example

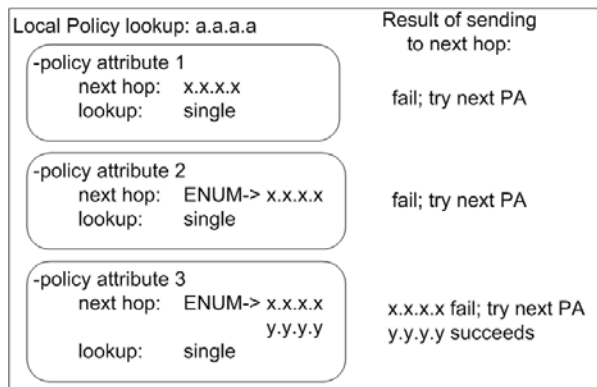
Multistage routing is enabled by setting a policy attribute's lookup parameter to multi. Instead of replacing the SIP message's request URI with the policy attribute's next hop address or response from an ENUM or LRT lookup, the system uses that next hop or ENUM or LRT lookup response to reconstruct the SIP message. The reconstructed SIP message is fed again through all configured local policy configuration elements (and policy attribute sub elements). Each time the Net-Net SBC re-evaluates a SIP message against local policies, it is considered an additional routing stage. When multiple records are returned from an ENUM or LRT lookup, the Net-Net SBC evaluates the first response against all applicable local policies. If unsuccessful, the Net-Net SBC evaluates all additional responses, in turn, against all applicable local policies.

For example:



Multistage Routing Example 2

The following three local policy configuration elements are configured in the Net-Net SBC:



The local route table in default-lrt appears as follows:

```
<route>
  <user type="E164">159</user>
    <next type="regex">!^.*$!sip:115680000000@192.168.200.47!</next>
    <next type="regex">!^.*$!sip:2156800000002@192.168.200.99!</next>
    <next type="regex">!^.*$!sip:115780000000@192.168.200.44!</next>
  </route>
```

1. The Net-Net SBC receives an INVITE on realm, private (SDP is omitted below):

```
INVITE sip:159@192.168.1.49:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.48:5060
From: sipp <sip:sipp@192.168.1.48:5060>;tag=1
To: sut <sip:159@192.168.1.49:5060>
Call-ID: 1-4576@192.168.1.48
CSeq: 1 INVITE
Contact: sip:sipp@192.168.1.48:5060
Max-Forwards: 70
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 135
```

2. The Net-Net SBC performs a local policy search based on the following parameters:

```
from-address: sipp <sip:sipp@192.168.1.48:5060>;tag=1
to-address: sip:159@192.168.1.49:5060
Source Realm: private
```

3. The local policy search returns the four following routes to try:

```
lrt:default-lrt
192.168.200.50
lrt:emergency
lrt:carrier-lrt
```

- a. The first next-hop route will be an LRT query. In addition, this policy attribute is configured with lookup=multi, meaning the results of the LRT query should be used for another local policy query, i.e., a second stage. More specifically, the request-uri that was received in response to the LRT query will be used as the to-uri in the next LP query.
- b. The Net-Net SBC performs the LRT lookup in the default-lrt configuration element and is returned the following:

```

sip:11568000000@192.168.200.47
sip:215680000002@192.168.200.99
sip:11578000000@192.168.200.44

```

c. The Net-Net SBC attempts to use the results from the LRT query for the next stage Local Policy lookup(s). Beginning with the first route and continuing in sequential order, the Net-Net SBC will try to route the outgoing INVITE message by performing additional Local Policy lookups on the remaining LRT query results, until the INVITE is successfully forwarded.

The Net-Net SBC performs a local policy query on:

```
sip:11568000000@192.168.200.47
```

Which equates to a local policy lookup on:

```

from-URI=sipp <sip:sipp@192.168.1.48:5060>;
to-URI=sip:11568000000@192.168.200.47
Source Realm: private

```

The query fails because there is no Local Policy entry for 11568000000.

d. The Net-Net SBC performs a second query on request-uri

```
sip:215680000002@192.168.200.99
```

Which equates to a local policy lookup on:

```

from-URI=sipp <sip:sipp@192.168.1.48:5060>;
to-URI=sip:215680000002@192.168.200.99
Source Realm: private

```

e. The LP query is successful and returns the following next- hops:

```

192.168.200.98
192.168.200.99
192.168.200.44

```

f. The three routes shown above represent the next stage of the multistage routing for this INVITE. The policy attributes' lookup parameter is set to single for these next-hops. Therefore, the SD will attempt to send the outgoing INVITE message to one or more of these next-hops; there are no more stages to check.

4. The Net-Net SBC sends an INVITE to 192.168.200.98:

```

INVITE sip:215680000002@192.168.200.98;lr SIP/2.0
Via: SIP/2.0/UDP 192.168.200.49:5060
From: sipp <sip:sipp@192.168.1.48:5060>
To: sut <sip:159@192.168.1.49:5060>
Call-ID: SDnhae701-76e8c8b6e168958e385365657faab5cb-v3000i1
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.200.49:5060;transport=udp>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140

```

5. If the INVITE is sent to 192.168.200.98 successfully, the local policy routing will conclude and the call will continue processing. Otherwise the SD will try the other next hops until a route succeeds or all next-hops have been exhausted

Customizing Lookup Keys

When the **next hop** parameter points to perform an ENUM or LRT lookup, it can be provisioned with a "key=" attribute in order to specify a parameter other than the

username to perform the lookup on. The following table lists the header, key value, and corresponding syntax to configure the Net-Net SBC with.

Username from Header:	Key Value	Example
To-URI	\$TO	key=\$TO
From-URI	\$FROM	key=\$FROM
P-Asserted-Identity	\$PAI	key=\$PAI

For a subsequent stage in multistage local policy routing, the lookup key to use for the next stage can be explicitly specified by configuring the **next key** parameter. By default, multistage lookups use the modified Request-URI returned from the ENUM/LRT response as the to-address key for the next local policy lookup. When the **next key** parameter is configured, its value will be used for the to-address key in the subsequent local policy lookup regardless if an ENUM or LRT lookup is configured for that policy attribute. The key syntax for this parameter is the same as with the Routing-based RN and CIC feature.

Multistage Routing Lookup Termination

It is important for the Net-Net SBC to have a mechanism to stop performing additional stages of route lookups and limit the number of attempts and results to be tried. Routing termination can be performed at in the non-multistage way or at the global session router level.

Global Local Policy Termination

The Net-Net SBC can be configured to limit local policy lookups based several aspects of the route lookup process:

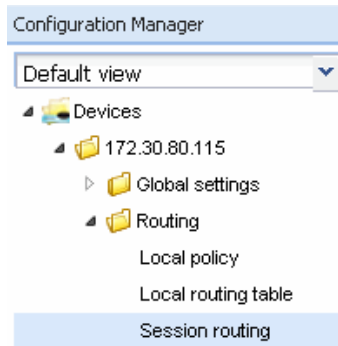
- Limiting the number of stages per message lookup—The Net-Net SBC can limit to the number of additional local policy lookup stages it will perform received message to a maximum of 5. This is configured with the **additional lp lookups** parameter. Leaving this parameter at its default value of 0 essentially disables multistaged local policy lookups.
- Limiting the number of routes per Local Policy lookup—The Net-Net SBC can limit the number of route results to use as returned for each Local-Policy lookup. This is configured with the **max lp lookups routes per lookup** parameter. Leaving this parameter at its default value of 0 places no limit on the number of returned routes the Net-Net SBC can try.
- Limiting the total number of routes for all local policy lookups per message request—The Net-Net SBC can limit the number of route returned in total across all lookups for a given request, including additional stages. This is configured with the **total lp routes** parameter. Leaving this parameter at its default value of 0 places no limit on the number of returned routes the Net-Net SBC can try. This parameter overrides any configured options.

Additionally, the Net-Net SBC monitors for local policy lookup loops which could cause a significant deterioration in performance. If a loop is found, the Net-Net SBC stops trying the looping route list and proceeds to try any remaining routes..

Configuring Policy Attributes

To configure policy attributes:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. **Number of additional lookup stages**—Enter the maximum number of additional local policy look-ups to be allowed for a given look-up request. You can enter a value between 0 and 5. The default value is 0 and it disables this feature.
5. **Maximum number of routes returned for each lookup**—Enter the maximum number of route results you want return for one local policy look-up. The default value for this parameter is 0, meaning there is no limit in place.
6. **Total number of policy local policy routes**—Enter the maximum number of routes returned across all local policy look-ups for a given request. The default value for this parameter is 0, meaning there is no limit in place.

Number of additional local policy lookups	5
Maximum number of routes returned for each lookup	10
Total number of local policy routes	250

7. Click **Apply**.

SIP Session Agent DNS-SRV Load Balancing

Previously, the Net-Net SBC provided the ability to specify an FQDN (fully qualified domain name) for a destination session-agent. During DNS lookup the FQDN could resolve to multiple SRV (Resource Record for Servers) records. Each SRV could resolve to a single IP address via A-Record query in IMS or DNS.

With Release 6.2.0 the Net-Net SBC supports load balancing behavior as described in RFC 3263, *Session Initiation Protocol (SIP): Locating SIP Servers*.

1. The Net-Net SBC will provide a new parameter **Ping all addresses** in session agent configuration mode to enable internal load balancing and RFC 3263 compliance. The Net-Net SBC monitor the availability of the dynamically resolved IP addresses obtained from DNS server using OPTIONS ping (ping-per-DNS entry). The ping method and ping interval for each resolved IP addresses will be copied from original session-agent.

Status of Session-Agent:

In Service – if any of dynamically resolved IP addresses is in service

Out of service – if all dynamically resolved IP addresses is out of service.

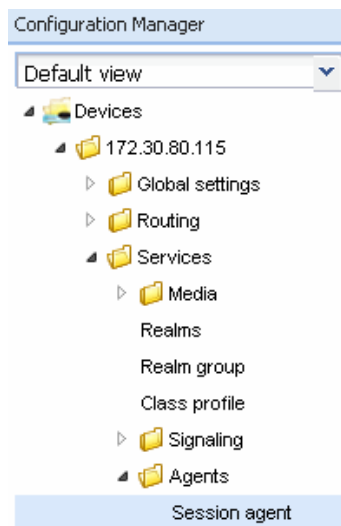
The default of **Ping all addresses** is **disabled**, in which case the Net-Net SBC only pings the first available resolved IP addresses.

2. With status of each resolved IP addresses above, the Net-Net SBC will recurse through the list of these in-service IP addresses dynamically resolved from DNS server on 503 response, and stop recursion based upon a configured list of response values specified by the **stop-recurse** parameter in SIP interface configuration mode. With internal load balancing enabled in the session-agent, the Net-Net SBC provides the ability to select routing destinations based on SRV weights. The priority/weight algorithm is based on RFC 2782, *A DNS RR for specifying the location of services (DNS SRV)*.
3. The Net-Net SBC will provide the similar functionality as that listed above for A-records, the Net-Net SBC will select first available routing destinations because there is no priority/weight contained in A-records.

Configuring Session Agent DNS-SRV Load Balancing

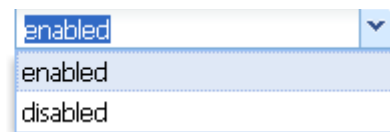
To configure session-agent DNS-SRV load balancing:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a SIP session agent you want to configure in the table and click **Edit**.
4. Click the Translation and manipulation tab at the top of the content area.
5. **Ping all addresses**—Select enabled from the combination box to enable session-agent DNS-SRV load balancing. The default state is **disabled**.

Ping all addresses



6. Click **Apply**.

Route (Local) Policy Matching for Parent Realms

For SIP and H.323, you can configure the Net-Net SBC to use the parent realm for routing purposes even when the source realm for an incoming message is a child realm.

With this feature disabled (default), the Net-Net SBC uses the specific source realm to perform a route policy look-up. When the source realm is a child realm and any relevant local policies are configured with the parent realm, there will be no matches and the route policy look-up will fail. To avoid this issue and ensure successful look-ups, you must configure multiple local policies if you want to use a configuration with nested realms.

The Net-Net SBC examines the source realm to determine if it is a parent realm with any child realms when you enable this feature. If the parent, source realm does have child realms, then the Net-Net SBC creates route policy entries for the parent and all of its child realms. This operation is transparent and can save time during the configuration process.

It is possible, then, for a route policy look-up to match the same child realm in two ways:

- Through a match via the parent realm
- Through a direct match for a route policy configured with that specific child realm

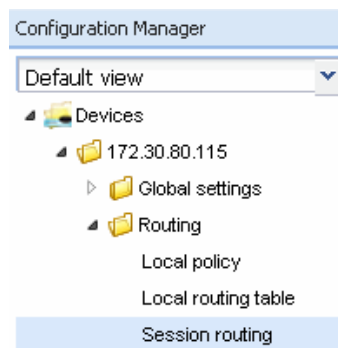
In such a case, the child realm must have different costs for each type of match to avoid collisions.

Configuring Route (Local) Policy Matching

This feature is enabled on a global basis in the session router configuration. Because it applies system-wide, all source realms will use this form of matching when enabled.

To enable route policy matching for parent realms:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. **Include parent realm in route policy matching**—Select enable from the combination box if you want the Net-Net SBC to perform route policy realm

matching based on the parent realm (so that there are route policy entries for parent and child realms).

Include parent realm in route policy
matching

5. Click **Apply**.

Source URI Information in ENUM Requests

ENUM queries can include the source URI that caused the ENUM request by enabling the **include source information in queries** parameter. The Net-Net SBC can add the P-Asserted-ID URI (only if not in an INVITE) or the From URI into an OPT-RR Additional Record to be sent to the ENUM server. It can be useful to specify the originating SIP or TEL URI from a SIP request which triggered the ENUM query, so the ENUM server can provide a customized response based on the caller.

This feature implements the functionality described in the Internet Draft, *DNS Extension for ENUM Source-URI*, draft-kaplan-enum-source-uri-00.

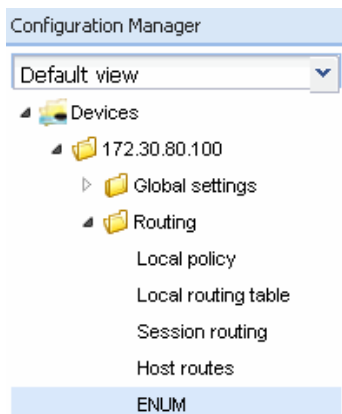
When a P-Asserted-ID is blocked or removed before the ENUM query is made, the Net-Net SBC only sends the URI in the From header.

Note that to support this feature, according to the Internet draft, ENUM clients must support 1220 bytes in UDP responses. Therefore, if this feature is enabled, and the maximum response size parameter is not set i.e., with a 512 byte default, the Net-Net SBC will set the size to 1200 on the OPT-RR records sent.

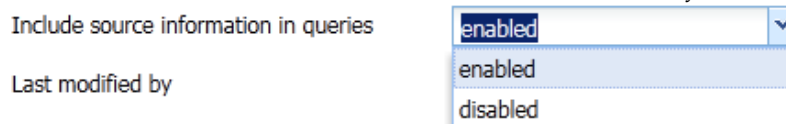
Configuring OPT-RR Additional Records Sent to ENUM Servers

To send OPT-RR additional records to ENUM servers:

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.
4. Click an ENUM configuration name in the ENUM configuration table and click **Edit**. The Edit ENUM configuration window appears.
5. **Include source information in queries**—Select enabled for the Net-Net SBC to send source URI information to the ENUM server with any ENUM queries.



6. Click **Apply**.

ENUM Failover and Query Distribution

The Net-Net SBC can intelligently distribute ENUM queries among all configured ENUM servers. By setting the **Query method** parameter to round-robin, the Net-Net SBC will cycle ENUM queries, sequentially, among all configured ENUM servers. For example, query 1 will be directed to server 1, query 2 will be directed to server 2, query 3 will be directed to server 3, and so on.

The default query method, hunt, directs all ENUM queries toward the first configured ENUM server. If the first server is unreachable, the Net-Net SBC directs all ENUM queries toward the next configured ENUM server, and so on.

Failover to New ENUM Configuration

When an ENUM's configured servers are unreachable via the network, for example, no response is received on a query, the Net-Net SBC can failover to a defined ENUM configuration that contains different ENUM servers to query. This failover behavior works when all servers in an ENUM configuration are unreachable, rather than when the Net-Net SBC receives not-found type responses.

The Net-Net SBC queries each ENUM server once before trying the next configured server, and then ultimately tries the servers listed in the **failover-to** ENUM

configuration. If the failover-to servers are also unreachable, the Net-Net SBC fails the call; the failover-to behavior does not recurse among ENUM configurations, it only checks the first, linked ENUM configuration.

ENUM Server Operation States

After five consecutive failed attempts, an ENUM server is considered Out of Service (OOS). All subsequent queries which would be directed to the OOS servers are immediately directed to the first non-OOS server. ENUM servers return to in-service after 600 seconds. If all configured ENUM servers are OOS, the Net-Net SBC fails the call.

After the first failed attempt to reach an ENUM server, it is placed in a time out state, which it stays in for 30 seconds. Within this 30 seconds it will not be contacted when an ENUM query is made. After the 30 seconds pass, the ENUM server goes back to an in-service state.

Server Availability Monitoring

The Net-Net SBC can probe an ENUM server's health by sending it a standard ENUM NAPTR query and receiving a valid answer. The query is for the phone number defined in the **health query number** parameter, which should be one that the ENUM servers can positively resolve. As long as the query succeeds, that ENUM server maintains its in-service state and is available for ENUM queries. Any lack of response, whether network based (time-outs), or application based (DNS error or "not found" response) is considered a query failure and the server is set to OOS and unavailable for ENUM queries.

The Net-Net SBC continuously checks the health of all configured ENUM servers to determine their current state and monitor for failed servers' return to service. All servers are checked for availability at the **health query interval** parameter, as defined in seconds.

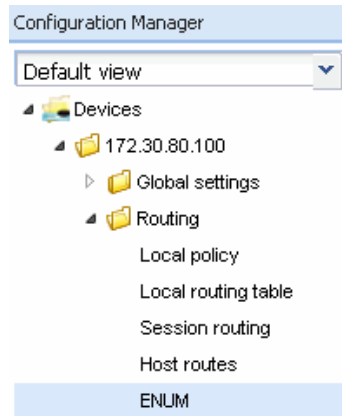
ENUM Server IP Address and Port

You can configure an IP address and port for each ENUM server listed in the **ENUM servers** parameter. IP address and port are specified in XXX.XXX.XXX.XXX:YYYY format with a port value range of 1024-65535. If the port number is not specified, 53 is assumed.

Configuring ENUM Failover and Query Distribution

To configure ENUM failover and query distribution:

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.
The Edit ENUM configuration window appears.
4. **ENUM server**—Click **Add**. The Add Enum server dialog box appears.
5. **Address**—Enter the IP address and port value of the ENUM servers (an ENUM server and corresponding redundant servers) to be queried.

 The 'Add Enum server' dialog box is shown. It has a text field labeled '*Address' containing '172.168.1.101'. To the right of the field is a note: '(Unique required. Enter a valid IP address)'. At the bottom are 'Apply' and 'Cancel' buttons.

6. Click **Apply**. The IP address is added to the ENUM servers list.
7. Repeat steps 5 through 7 to add additional ENUM servers.
8. **Query method**—Click the strategy the Net-Net SBC uses to contact ENUM servers in the drop down list. The valid values are:
 - **hunt**: Directs all ENUM queries toward the first configured ENUM server. If the first server is unreachable, the Net-Net SBC directs all ENUM queries toward the next configured ENUM server, and so on.
 - **round-robin**: Cycles all ENUM queries, sequentially, among all configured ENUM servers. Query 1 will be directed to server 1, query 2 will be directed to server 2, query 3 will be directed to server 3.

 The 'Query method' dropdown menu is shown. The current selection is 'hunt'. The dropdown list is open, showing 'hunt' and 'round-robin' as options.

9. **Health query number**—Enter a standard ENUM NAPTR query that will consistently return a positive response from the ENUM server.

 The 'Health query number' text field is shown with the value '18885551111' entered.

10. **Health query interval (sec)**—Enter the number of seconds to perpetually probe ENUM servers for health. The valid values are **0** through **65535** seconds. The default value is **0**.

Health query interval (secs)

0

11. **Failover ENUM configuration**—Enter the name of another ENUM configuration in the text box to failover to under appropriate conditions.

Failover to

Config2

12. Click **Apply**.

Source URI Information in ENUM Requests

ENUM queries can include the source URI that caused the ENUM request by enabling the **include source information in queries** parameter. The Net-Net SBC can add the P-Asserted-ID URI (only if not in an INVITE) or the From URI into an OPT-RR Additional Record to be sent to the ENUM server. It can be useful to specify the originating SIP or TEL URI from a SIP request which triggered the ENUM query, so the ENUM server can provide a customized response based on the caller.

This feature implements the functionality described in the Internet Draft, *DNS Extension for ENUM Source-URI*, draft-kaplan-enum-source-uri-00.

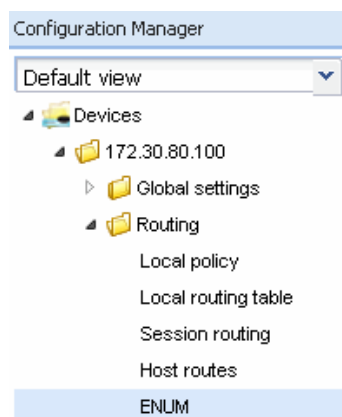
When a P-Asserted-ID is blocked or removed before the ENUM query is made, the Net-Net SBC only sends the URI in the From header.

Note that to support this feature, according to the Internet draft, ENUM clients must support 1220 bytes in UDP responses. Therefore, if this feature is enabled, and the maximum response size parameter is not set i.e., with a 512 byte default, the Net-Net SBC will set the size to 1200 on the OPT-RR records sent.

Configuring OPT-RR Additional Records Sent to ENUM Servers

To send OPT-RR additional records to ENUM servers:

1. Expand the Configuration Manager slider > Routing.



2. Click ENUM. The ENUM configuration table opens in the content area.
3. Select an ENUM configuration and click **Edit**.

4. Click an ENUM configuration name in the ENUM configuration table and click **Edit**. The Edit ENUM configuration window appears.
5. **Include source information in queries**—Select enabled for the Net-Net SBC to send source URI information to the ENUM server with any ENUM queries.

Include source information in queries

Last modified by

6. Click **Apply**.

Local Route Tables (Onboard ENUM)

Adding onto the Net-Net SD's existing ENUM capabilities, the Local Route Tables feature gives the Net-Net SD the ability to function as an ENUM server and to transform E.164 numbers to URIs locally. This ability provides extensive flexibility for routing.

This feature introduces the concept of a “local route cache,” which is similar to an ENUM cache. The local route cache is populated by a local file on the Net-Net SD, a file that specifies numbers, prefixes, and their next-hops. For routing, the local route cache operates in a way similar to the ENUM model in that a local policy next hop specifies the local route table look-up.

In a peering deployment, an NCS call agent and SIP registrar forward SIP requests to the Net-Net SD, which is acting in proxy mode. These requests contain Requests URIs that might have—along with TN (telephone number)—a routing number (RN) or carrier identification code (CIC) user-parameter. The chosen identifier (TN, RN, or CIC) is used to perform an ENUM lookup, and then the Net-Net SD updates the Request URI with the resulting NAPTR regex field. The new hostname portion is used to identify the next hop. That next hop might be a session agent, or the Net-Net SD might use DNS to find the next hop—hence the needs for the Net-Net SD to store the equivalent data and perform routing based on it as if an ENUM request had been performed.

How It Works

The Net-Net SD supports a new system task for local routing, and has the ability to perform an ENUM lookup for SIP requests. It handles ENUM requests (NAPTR requests) from the system's SIP task and is responsible for both for performing the ENUM lookup and communicating the results to the SIP task. The new task is also responsible for processing the new local routing configuration objects.

When a SIP call is being routed, the Net-Net SD uses local policy attributes to determine if a local ENUM query (local route table lookup) is required and, if one is needed, which local routing configuration to use. Successful local ENUM queries result in URIs that can be used to continue routing or redirecting calls.

ENUM Lookup Key and Length

When the Net-Net SD performs ENUM queries to a DNS server based on TNs, the TN is the normalized user part of the Request URI with any of the user parameters stripped from it. Normalizing a TN involves stripping out the non-digit characters, but leaving the plus sign (+) if it exists. Using local route tables means that you can set a flag for the next-hop parameter for a set of local policy attributes that specifies the user parameter should be used as the key for an ENUM query. Commonly, the

user parameter is RN, but you can set it to any value you require. If you provide a key that does not exist, the query/route will fail and the Net-Net SD will attempt next route or fail to send an appropriate response.

In addition you can also configure a lookup length in the ENUM configuration that provides for more efficient caching of URI lookup results; in it, you can specify the length of the string for the DNS request starting from the most significant digit. This also provides more flexibility for length matching is useful given the amount of wild card matching available in ENUM services. Specific ENUM groups might only be intended to provide NPANXX or wild card results.

Net-Net Central Instructions and Examples

This section shows you how to:

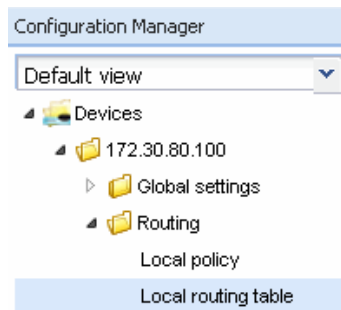
- Set up local route configuration
- Specify that a set of local policy attributes needs to use local routing
- Specify that a set of local policy attributes should use an RN for lookup

Setting Up a Local Routing Configuration

You configure a name for the local ENUM server, the filename you want to give to the database corresponding to this server, and the prefix length (significant digits/bits) to be used for lookup.

To set up a local routing configuration:

1. Expand the Configuration Manager slider > Routing.



2. Click Local routing table. The local routing table appears in the content area.
3. Click **Add**. The Add Local routing table dialog box appears.
4. **Name**—Enter a unique identifier for the local ENUM server; this name is used for reference in the local policy attributes when to specify that local routing should be used.

*Name

5. **File name**—Enter the name for the file from which the database corresponding to this ENUM server will be created. You should use the .gz format, and the file should be placed in the /code/lrt/ directory of the Net-Net SD.

*File name

6. Click **Apply**.
7. Select the newly created local routing configuration and click **Edit**.

8. **Prefix length**—Enter the number of significant digits/bits to used for lookup and cache storage. The default is 0.

Prefix length

0

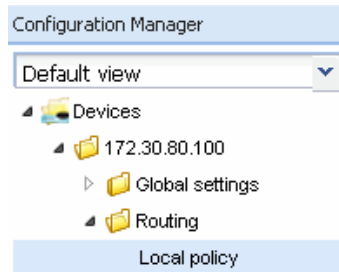
9. Click **Apply**.

Applying the Local Routing Configuration

The way that you apply the local routing configuration is to call it to use in the local policy attributes. You do this by setting a flag in the next-hop parameter along with the name of the local routing configuration that you want to use.

To apply the local routing configuration:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Click a route in the table to select it and click **Edit**. The Edit Routes window appears.
4. Scroll to the Route policies section.
5. In the Route policies table, click a policy to select it and click **Edit**. The Edit Route policy window appears.
6. Scroll to the Next hop section.
7. **Next hop**—Enter lrt:lookup in the text box to tell the Net-Net SD that a local route table will be used.

*Next hop

lrt:lookup

You can also set the lookup key to an RN in the local policy attributes' next-hop parameter by appending ;key=rn to lrt:lookup.

*Next hop

lrt:lookup;key=rn

8. Click **Apply**. You return to the Edit Routes window.
9. Click **Apply**. You return to the Routes table.

Multi-Stage Local Policy Routing for SIP

Multistage local policy routing enables the Net-Net SBC to perform multiple stages of route lookups where the result from one stage is used as the lookup key for the next routing stage.

Routing Stages

A routing stage signifies a re-evaluation of local policy based on the results of a local policy lookup. In the simplest, single stage case, the Net-Net SBC performs a local policy lookup on a SIP message's Request URI. The result of that local policy lookup is a next hop FQDN, IP address, ENUM lookup, or LRT lookup; that result is where the Net-Net SBC forwards the message. In the multistage routing model, that resultant next hop is used as the lookup key for a second local policy lookup.

The results of each stage do not erase the results of the previous stage. Thus, previous results are also possible routes to use for recursion, but the next stage results are tried first.

Note: Setting a next hop to a SAG in a multistage scenario constitutes an error.

Network Applications

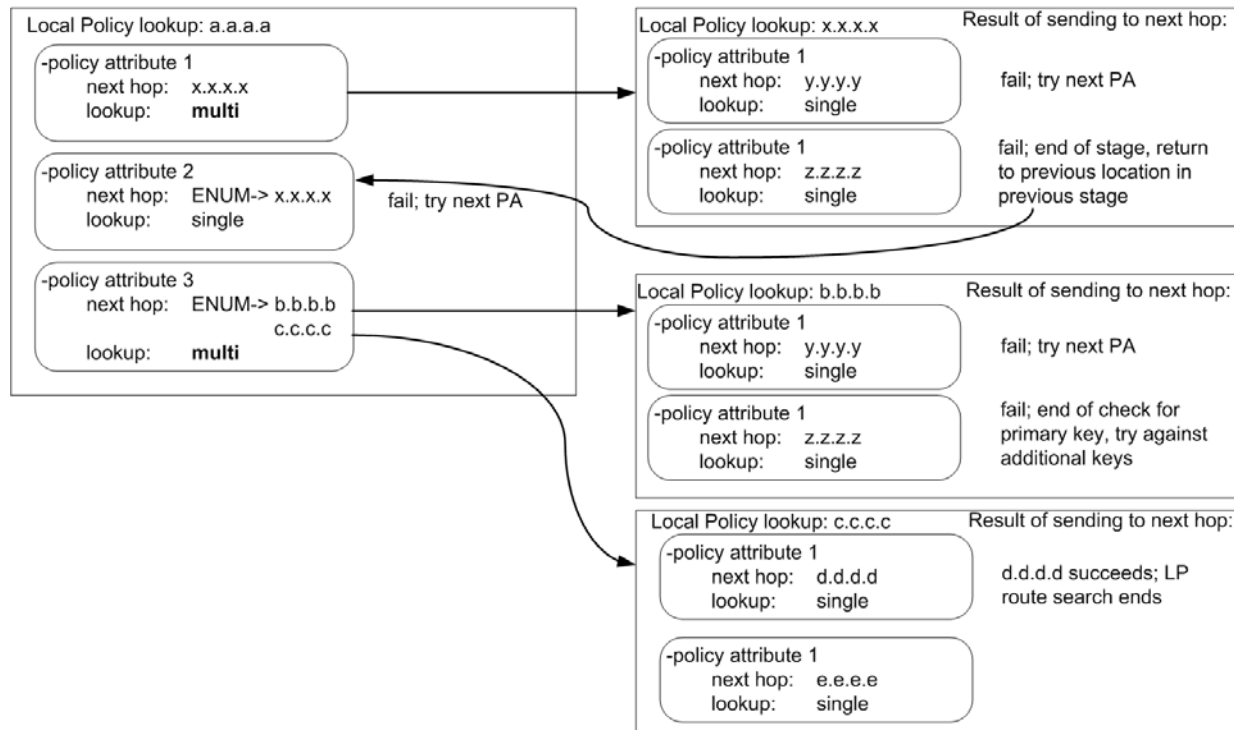
In the following typical applications of multistage routing, an operator might need to query:

- An ENUM server for a destination number. Based on the NAPTR result of the ENUM query, the Net-Net SBC performs a local policy lookup to decide how to route the request, perhaps based on an LRT table lookup.
- One ENUM server for a number portability lookup, then based on the routing number perform a second ENUM query to a different server to learn which carrier to use for the routing number. Then, then based on the identified carrier perform an LRT lookup for what next-hop(s) to use for that carrier.
- An LRT table to confirm the allowed source number. Then, based on the result, query an ENUM server for destination routing.

Multistage Routing Conceptual Example

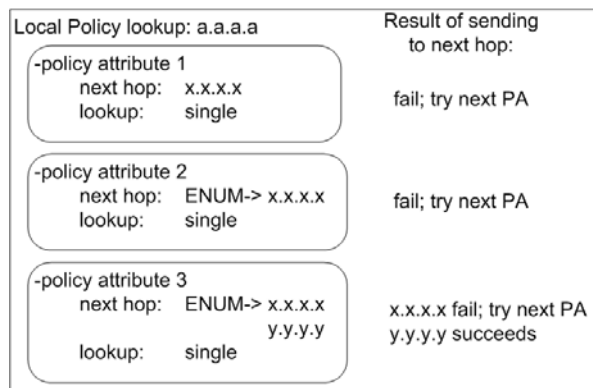
Multistage routing is enabled by setting a policy attribute's lookup parameter to multi. Instead of replacing the SIP message's request URI with the policy attribute's next hop address or response from an ENUM or LRT lookup, the system uses that next hop or ENUM or LRT lookup response to reconstruct the SIP message. The reconstructed SIP message is fed again through all configured local policy configuration elements (and policy attribute sub elements). Each time the Net-Net SBC re-evaluates a SIP message against local policies, it is considered an additional routing stage. When multiple records are returned from an ENUM or LRT lookup, the Net-Net SBC evaluates the first response against all applicable local policies. If unsuccessful, the Net-Net SBC evaluates all additional responses, in turn, against all applicable local policies.

For example:



Multistage Routing Process Example

The following three local policy configuration elements are configured in the Net-Net SBC:



The local route table in default-lrt appears as follows:

```
<route>
  <user type="E164">159</user>
  <next type="regex">!^.*$!sip:11568000000@192.168.200.47!</next>
  <next type="regex">!^.*$!sip:215680000002@192.168.200.99!</next>
  <next type="regex">!^.*$!sip:11578000000@192.168.200.44!</next>
</route>
```

1. The Net-Net SBC receives an INVITE on realm, private (SDP is omitted below):

```
INVITE sip:159@192.168.1.49:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.48:5060
```

```

From: sipp <sip:sipp@192.168.1.48:5060>;tag=1
To: sut <sip:159@192.168.1.49:5060>
Call-ID: 1-4576@192.168.1.48
CSeq: 1 INVITE
Contact: sip:sipp@192.168.1.48:5060
Max-Forwards: 70
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 135

```

2. The Net-Net SBC performs a local policy search based on the following parameters:

```

from-address: sipp <sip:sipp@192.168.1.48:5060>;tag=1
to-address: sip:159@192.168.1.49:5060
Source Realm: private

```

3. The local policy search returns the four following routes to try:

```

lrt:default-lrt
192.168.200.50
lrt:emergency
lrt:carrier-lrt

```

- a. The first next-hop route will be an LRT query. In addition, this policy attribute is configured with lookup=multi, meaning the results of the LRT query should be used for another local policy query, i.e., a second stage. More specifically, the request-uri that was received in response to the LRT query will be used as the to-uri in the next LP query.

- b. The Net-Net SBC performs the LRT lookup in the default-lrt configuration element and is returned the following:

```

sip:11568000000@192.168.200.47
sip:215680000002@192.168.200.99
sip:11578000000@192.168.200.44

```

- c. The Net-Net SBC attempts to use the results from the LRT query for the next stage Local Policy lookup(s). Beginning with the first route and continuing in sequential order, the Net-Net SBC will try to route the outgoing INVITE message by performing additional Local Policy lookups on the remaining LRT query results, until the INVITE is successfully forwarded.

The Net-Net SBC performs a local policy query on:

```
sip:11568000000@192.168.200.47
```

Which equates to a local policy lookup on:

```

from-URI=sipp <sip:sipp@192.168.1.48:5060>;
to-URI=sip:11568000000@192.168.200.47
Source Realm: private

```

The query fails because there is no Local Policy entry for 11568000000.

- d. The Net-Net SBC performs a second query on request-uri

```
sip:215680000002@192.168.200.99
```

Which equates to a local policy lookup on:

```

from-URI=sipp <sip:sipp@192.168.1.48:5060>;
to-URI=sip:215680000002@192.168.200.99
Source Realm: private

```

- e. The LP query is successful and returns the following next- hops:

```
192.168.200.98
```

192.168.200.99

192.168.200.44

f. The three routes shown above represent the next stage of the multistage routing for this INVITE. The policy attributes' lookup parameter is set to single for these next-hops. Therefore, the SD will attempt to send the outgoing INVITE message to one or more of these next-hops; there are no more stages to check.

4. The Net-Net SBC sends an INVITE to 192.168.200.98:

```
INVITE sip:215680000002@192.168.200.98;lr SIP/2.0
Via: SIP/2.0/UDP 192.168.200.49:5060
From: sipp <sip:sipp@192.168.1.48:5060>
To: sut <sip:159@192.168.1.49:5060>
Call-ID: SDnhae701-76e8c8b6e168958e385365657faab5cb-v3000i1
CSeq: 1 INVITE
Contact: <sip:sipp@192.168.200.49:5060;transport=udp>
Max-Forwards: 69
Subject: Performance Test
Content-Type: application/sdp
Content-Length: 140
```

5. If the INVITE is sent to 192.168.200.98 successfully, the local policy routing will conclude and the call will continue processing. Otherwise the SD will try the other next hops until a route succeeds or all next-hops have been exhausted.

Customizing Lookup Keys

When the **next hop** parameter points to perform an ENUM or LRT lookup, it can be provisioned with a "key=" attribute in order to specify a parameter other than the username to perform the lookup on. The following table lists the header, key value, and corresponding syntax to configure the Net-Net SBC with.

Username from Header:	Key Value	Example
To-URI	\$TO	key=\$TO
From-URI	\$FROM	key=\$FROM
P-Asserted-Identity	\$PAI	key=\$PAI

For a subsequent stage in multistage local policy routing, the lookup key to use for the next stage can be explicitly specified by configuring the **next key** parameter. By default, multistage lookups use the modified Request-URI returned from the ENUM/LRT response as the to-address key for the next local policy lookup. When the **next lookup key** parameter is configured, its value will be used for the to-address key in the subsequent local policy lookup regardless if an ENUM or LRT lookup is configured for that policy attribute. The key syntax for this parameter is the same as with the Routing-based RN and CIC feature.

Multistage Routing Lookup Termination

It is important for the Net-Net SBC to have a mechanism to stop performing additional stages of route lookups and limit the number of attempts and results to be tried. Routing termination can be performed at in the non-multistage way or at the global session router level.

Global Local Policy Termination

The Net-Net SBC can be configured to limit local policy lookups based several aspects of the route lookup process:

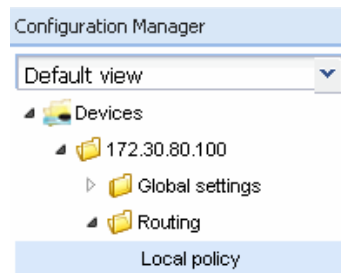
- Limiting the number of stages per message lookup—The Net-Net SBC can limit to the number of additional local policy lookup stages it will perform received message to a maximum of 5. This is configured with the **number of additional lookup stages** parameter. Leaving this parameter at its default value of 0 essentially disables multistaged local policy lookups.
- Limiting the number of routes per Local Policy lookup—The Net-Net SBC can limit the number of route results to use as returned for each Local-Policy lookup. This is configured with the **max lp lookups routes per lookup** parameter. Leaving this parameter at its default value of 0 places no limit on the number of returned routes the Net-Net SBC can try.
- Limiting the total number of routes for all local policy lookups per message request—The Net-Net SBC can limit the number of route returned in total across all lookups for a given request, including additional stages. This is configured with the **total lp routes** parameter. Leaving this parameter at its default value of 0 places no limit on the number of returned routes the Net-Net SBC can try. This parameter overrides any configured options.

Additionally, the Net-Net SBC monitors for local policy lookup loops which could cause a significant deterioration in performance. If a loop is found, the Net-Net SBC stops trying the looping route list and proceeds to try any remaining routes.

Configuring Multi-Stage Local Policy Routing

To configure multi-stage local policy routing:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Click the local policy you want to configure in the table and click **Edit**.
4. Scroll to Route policy attributes.
5. Click the route policy attribute you want to configure and click **Edit**. The Edit Route policy window appears.
6. **Lookup**—Click the local policy look-up strategy you want to use for this set of policy attributes in the drop down list. The default is **single**.



7. **Next key**—Enter the key you want to use for the next stage of local policy look-up. You can use the pre-defined values:
 - \$TO: To-URI username and host
 - \$PAI: P-Asserted-Identity URI username and host

This parameter has no default and is not required.

Next key

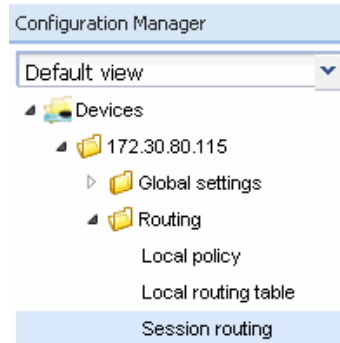
\$PAI

8. Click **Apply**.

Configuring Policy Attributes

To configure policy attributes:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. **Number of additional local policy lookups**—Enter the maximum number of additional local policy look-ups to be allowed for a given look-up request. You can enter a value between 0 and 5. The default value is 0 and it disables this feature.

Number of additional local policy lookups

0

5. **Maximum number of routes returned for each lookup**—Enter the maximum number of route results you want return for one local policy look-up. The default value for this parameter is 0, meaning there is no limit in place.

Maximum number of routes returned for each lookup

0

6. **Total number of local policy routes**—Enter the maximum number of routes returned across all local policy look-ups for a given request. The default value for this parameter is 0, meaning there is no limit in place.

Total number of local policy routes

0

7. Click **Apply**.

Unique Header Manipulation Rule Regex Patterns

In addition to the Header Manipulation Rule (HMR) support it offers, the Net-Net SBC can now be provisioned with unique regex patterns for each logical remote entity. Unique Regex patterns saves you provisioning time and saves Net-Net SBC resources in instances when it was previously necessary to a unique SIP manipulation per PBX for a small number of customer-specific rules.

Manipulation Pattern Per Remote Entity

On the Net-Net SBC, you can configure logical remote entities (session agents, realms, and SIP interfaces) with a manipulation pattern string that the system uses as a regular expression. Then the SIP manipulation references this regular expression using the reserved word `$MANIP_PATTERN`. At runtime, the Net-Net SBC looks for the logical entity configured with a manipulation pattern string in this order of preference: session agent, realm, and finally SIP interface.

On finding the logical entity configured with the manipulation string, the Net-Net SBC dynamically determines the expression. This makes it possible for the same SIP manipulation to make a valid reference to a `$MANIP_PATTERN` for one call but not for the next. When there is an invalid reference to a manipulation pattern (either dynamically or by configuration error), the pattern-rule expression that results will turn out to be the default expression (which is `\, +`).

When the `$MANIP_PATTERN` is used in a manipulation rule's **new value** parameter, it resolves to an empty string, equivalent of no value. Even though this process ends with no value, it still consumes system resources. And so Acme Packet recommends you do not use `$MANIP_PATTERN` as a **new value** value.

Reject Action

When you use the new action type called **reject** to all manipulation rules and a condition matching the manipulation rule arises, the Net-Net SBC rejects the request (though it does not drop responses) and increments a counter.

- If the **Message type** parameter is set to **any** and the message is a response, the Net-Net SBC increments a counter to show the intention to reject the message—but the message will continue to be processed.
- If the **Message type** parameter is set to **any** and the message is a request, the Net-Net SBC performs the rejection and increments the counter.

The **New value** parameter supplies the status code and reason phrase corresponding to the reject. You can use the following syntax to supply this information: `status-code[:reason-phrase]`. You do not have to supply the status code and reason phrase information; by default, the system uses `400:Bad Request`.

If you do supply this information, then the status code must be a positive integer between 300 and 699. The Net-Net SBC then provides the reason phrase corresponding to the status code. And if there is no reason phrase, the system uses the one for the applicable reason class.

You can also customize a reason phrase. To do so, you enter the status code followed by a colon (:), being sure to enclose the entire entry in quotation marks. ("`400:Bad Request`").

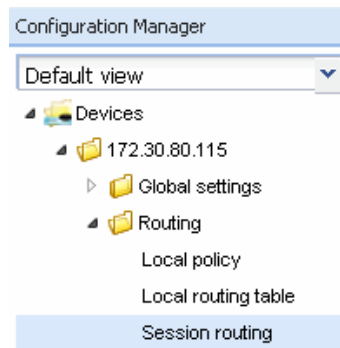
When the Net-Net SBC performs the **reject** action, the current SIP manipulation stops processing and does not act on any of the rules following the **reject** rule. This course of action is true for nested SIP manipulations that might have been constructed using the **sip-manip** action type.

Configuring the Reject Message Number and Time Window

To support the **reject** action, two parameters allow you to set how many messages in a certain amount of time cause the Net-Net SBC to generate an SNMP trap.

To configure the reject message number and time window:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. Scroll to the SIP manipulation counter section.
5. **Minimum reject message threshold to trigger trap**—Enter the minimum number of message rejections allowed in the **reject message window** time on the Net-Net SBC (when using the SIP manipulation action reject) before generating an SNMP trap. The default is 0, meaning this feature is disabled and no trap will be sent.
6. **Size of the reject message window (sec)**—Enter the time in seconds that defines the window for maximum message rejections allowed before generating an SNMP trap. The default is 0, meaning this feature is disabled and no trap will be sent.

Minimum reject message threshold to trigger trap	0
Size of the reject message window (sec)	10

7. Click **Apply**.

Log Action

When you use the action type called **log** and a condition matching the manipulation rule arises, the Net-Net SBC logs information about the current message to a separate log file. This log file will be located on the same core in which the SIP manipulation occurred, in the `sip` process. On the core where `sip` runs, a log file called `matched.log` will appear when this action type is executed.

The `matched.log` file contains a timestamp, received and sent Net-Net SBC network interface, sent or received IP address:port information, and the peer IP address:port information. It also specifies the rule that triggered the log action in this syntax: `rule-type[rule:name]`. The request URI, Contact header, To Header, and From header are also present.

```
-----
Apr 17 14:17:54.526 On [0:0]192.168.1.84:5060 sent to 192.168.1.60:5060
element-rule[checkRURIPort]
INVITE sip:service@192.168.1.84:5060 SIP/2.0
From: sipp <sip:+2125551212@192.168.1.60:5060>;tag=3035SIPpTag001
To: sut <sip:service@192.168.1.84>
```


Contact: sip:sipp@192.168.1.60:5060

Changes to Storing Pattern Rule Values

Release D7.0.0 introduces changes to the framework for storing regular expression results within manipulation rules, altering the way the **store** action works. These changes are beneficial to performance.

In previous releases, when the **store** action is used, the Net-Net SBC stores all values matching the regular expression defined in the **match-value** parameter for all headers. At runtime, the system evaluates all stored values to find the correct index. And any could reference another rule's stored value.

Now, you no longer need to specify the **store** action. The simple fact of referencing another rule tells the system it must store a value. When SIP manipulation is used, the system first checks to see if any values require storing. The **add** action is an exception to this process; storing happens after a header is added.

Removal of Restrictions

The following restrictions related to HMR have been removed in Release D7.0.0:

- The action **find-replace-all** now executes all element rules. Previously, no child rules were executed.
- The action **sip-manip** now executes existing all element rules. Previously, no child rules were executed.
- The action **store** now executes existing all element rules. Previously, only child rules with the **store** action were executed.
- The action **add** now executes existing all element rules. Previously, only child rules with the **add** action were executed.

Name Restrictions for Manipulation Rules

Historically, you have been allowed to configure any value in the following parameters:

- Header rules
- Element rules
- Mime rules
- Mime header rules
- Mime Isup rules
- ISUP param rules

This method of naming caused confusion when referencing rules, so now manipulation rules name must follow a specific syntax. They must match the expression `“^[[:alpha:]]*[:alnum: ]+$”` and contain at least one lower case letter.

In other words, the name must:

- Start with a letter, and then it can contain any number of letters, numbers, or underscores
- Contain at least one lower case letter

All pre-existing configurations will continue to function normally. If you want to change a manipulation rule, however, you are required to change its name.

New Value Restrictions

To simplify configuration and remove possible ambiguity, the use of boolean and equality operators (==, <=, <, etc.) for **new value** parameter values has been banned. Since there was no specific functionality tied to their use, their ceasing to be use will have no impact to normal SIP manipulation operations.

LRT Entry Matching

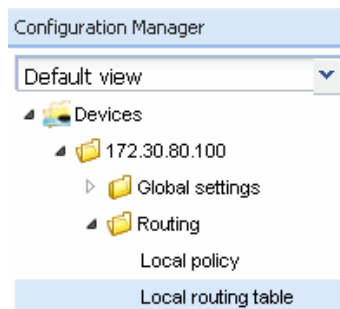
When searching an LRT for a matching route, the Net-Net SBC can be configured with one of three match modes with the *match mode* parameter in the *local routing config*. These modes are:

- **exact**—When searching the applicable LRT, the search and table keys must be an exact match.
- **best**—The longest matching table key in the LRT is the chosen match.
- **all**—The all mode makes partial matches where the table's key value is a prefix of the lookup key. For example, a lookup in the following table with a key of 123456 returns entries 1, 3, and 4. The 'all' mode incurs a performance penalty because it performs multiple searches of the table with continually shortened lookup keys to find all matching entries. This mode also returns any exact matches too.

Entry#	Key	Result
1	1	<sip:\0@host1.example.com>
2	122	<sip:\0@host22.example.com>
3	123	<sip:\0@host3.example.com>
4	1234	<sip:\0@host4.example.com>
5	1234567	<sip:\0@host7.example.com>
6	1235	<sip:\0@host5.example.com>

Instructions

1. Expand the Configuration Manager slider > Routing.



2. Click Local routing table. The local routing table appears in the content area.
3. **Match mode**—Select either **best**, **all**, or leave it as **exact** which is the default. This indicates to the Net-Net SBC how to determine LRT lookup matches.
4. Click **Apply**.

2349- LRT String Lookup

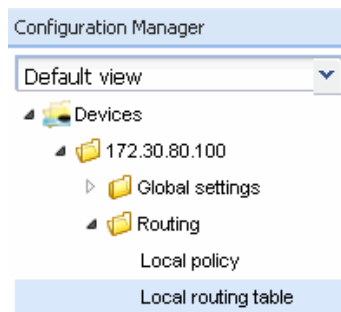
The Net-Net SBC can search an LRT for either E.164 or string table keys. This selection is on a global basis. When the string-lookup parameter is **disabled** (default) in the *local routing configuration*, all lookups will be E.164 type, except when:

- If *eloc-str-lookup* is enabled in a matching *local policy's policy-attribute*, E-CSCF procedures are applied and the resulting lookup type is 'string'.
- The Net-Net SBC also performs string lookups exclusively when a compound lookup key is specified.

When the lookup type is 'E.164', the lookup is skipped if the lookup key is not a valid telephone number (i.e. it must contain only digits).

Instructions

1. Expand the Configuration Manager slider > Routing.



2. Click Local routing table. The local routing table appears in the content area.
3. **String lookup**—Select **enabled** for the Net-Net SBC to perform LRT lookups on table keys of a string data type. Leave this parameter to its default as disabled to continue using E.164 type lookups.
4. Click **Apply**.

Directed Egress Realm from LRT/ENUM - 2473

A message can be sent into a specific egress realm identified in an ENUM query or LRT lookup. The egress realm is noted by a configurable parameter in the result URI. The Net-Net SBC is configured with the name of this parameter, that indicates an egress realm name, and looks for it in the returned URI.

To configure the parameter name, the *egress-realm-param* option is added to the *sip config* and/or the *h323 config* using the following format:

```
egress-realm-param=<name>
```

Where <name> is the parameter name to extract the egress realm name from.

When the *egress realm param* is defined, the ENUM and LRT results will always be checked for the presence of the URI parameter. The *sip config* options will apply for received SIP requests. The *h323 config* option will apply for received H.323 messages.

For example, if **egress-realm-param=egress** is added to the sip config, a matching entry in the LRT that specifies the egress realm **core** will look like this:

```
<route>
<user type="E164">+17815551212</user>
```

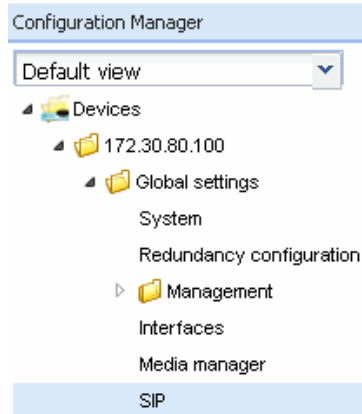
```
<next type="regex">!^.*$!sip:\00@core.example.com;egress=core!</next>
</route>
```

If the URI does not contain the parameter or the parameter identifies a realm that is not configured on the system, the egress realm that is normally applicable (from local policy, SIP-NAT, or session-agent data) will be used.

Instructions

To add an egress parameter to look for in a sip-config:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to Options and click **Options**.
5. In the User defined entry field, type the parameter **egress-realm-param** with the argument **egress** to parse for a returned ENUM or LRT result. For example:
 - **egress-realm-param=egress**

In order to append the new option to the sip-config's options list, you must prepend the new option with a "plus" sign. For example:

+egress-realm-param=egress

6. Click **Apply**.

Note: The egress-realm-param option can be configured similarly in the h323-config.

SIP Embedded Route Header

The Net-Net SBC examines the ENUM and LRT lookup result for embedded Route headers. In the LRT or as returned in an ENUM query a URI including an embedded route header would look like:

```
<sip:user@example.com?Route=%3Csip:host.example.com;lr%3E>
```

Using embedded Route headers is the Net-Net SBC's default behavior. This can be overridden by adding the *sip-config* option *use-embedded-route*.

When the ENUM or LRT result becomes the top Route header, any embedded Route headers extracted are inserted just after that top Route (which will always be a loose

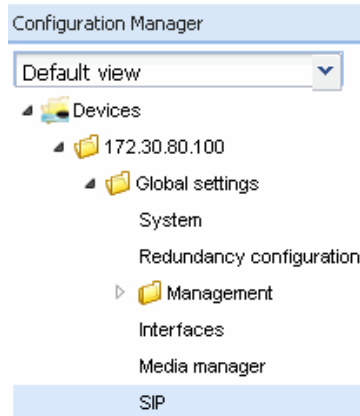
route and include the "lr" URI parameter). In this case, the request will be sent to the top Route.

When the ENUM or LRT results become the Request-URI, any embedded Route headers extracted from the result are inserted before any Route headers already in the outgoing request. After that, if there are any Route headers in the outgoing request and the top Route header has an "lr" URI parameter, the request is sent to the top Route header. Otherwise, the request is sent to the Request URI.

Instructions

To set the Net-Net SBC's default behavior of using embedded route headers from ENUM queries or LRT lookups:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Scroll to Options and click **Options**.
5. In the User defined entry field, type the parameter **use-embedded-route** with one of the following arguments:
 - **all**= use embedded routes from both ENUM and LRT results (default)
 - **none**= do not use embedded routes
 - **enum**= use embedded routes from ENUM results only
 - **lrt**= use embedded routes from LRT results only

In order to append the new option to the sip-config's options list, you must prepend the new option with a "plus" sign. For example:
6. Click **Apply**.

LRT Lookup Key Creation

Arbitrary LRT Lookup Key

In addition to the standard From, To, and P-Asserted-Identity header fields the Net-Net SBC can now use the values from any arbitrary SIP header as an LRT or ENUM lookup key. This is preformed by prepending a dollar sign "\$" by the header name whose value's userinfo portion of the URI will be used as the lookup key. For example, **key=\$Refer-To** would use the userinfo portion of the URI in the Refer-To header of the request as the lookup key.

An ampersand "&" followed by a header name will use the whole value of the header as the lookup key. For example, **key=&X-Route-Key** would use the whole value of the X-Route-Key as the lookup key. As a shortcut, an ampersand is not required for a "hidden" header. For example, "key=@LRT-Key" would use the value of the @LRT-Key header as the lookup key.

Hidden Headers for HMR and LRT lookup

When an LRT lookup key is more complex than just the URI's userinfo or a Tel-URI, HMR can be used to extract the data and build a special header.

By using a header name that begins with the at-sign "@" (e.g. @lrt-key), the header can be hidden and not included in outgoing SIP message, thus eliminating the need for an extra HMR rule to remove it.

Since '@' is not a valid character in a header name as defined by RFC 3261, there is no possibility of a collision between a header name defined in the future and a hidden header name beginning with "@".

Compound Key LRT Lookup - 2447

ENUM or LRT lookup keys can be combinations of more than one key value. For example, "key=\$FROM,\$TO" would construct a compound key with the userinfo of the From URI followed by a comma flowed by the userinfo of the To URI.

If the request message contained:

```
From: <sip:1234@example.com>
To: <sip:5678@example.com>
```

The compound key to match this From/To pair would be "1234,5678".

In the table lookup, the compound key is a single key value and there is no special treatment of the comma in key matching. It is simply an ordinary additional character that is matched like any letter or digit (i.e. the comma must appear in the LRT entry's "type" element data). For example, if the table were configured for "best" match-mode, the lookup key "1234,5678" would match a table entry of "1234,567", but it would not match a table entry of "123,5678".

2274 - Retargeting LRT/ENUM-based Requests

Request re-targeting is when a target or a request as indicated in the Request-URI, is replaced with a new URI.

The happens most commonly when the "home" proxy of the target user replaces the Request-URI with the registered contact of that user. For example, the original request is targeted at the Address-of-Record of bob (e.g. sip: bob@example.net). The "home" proxy for the domain of the original target, example.net, accesses the location service/registration database to determine the registered contact(s) for the user (e.g. sip:bob@192.168.0.10). This contact was retrieved in a REGISTER request

from the user's UA. The incoming request is then re-targeted to the registered contact. When *retarget-requests* is **enabled**, or the original Request-URI is the Net-Net SBC itself, the URI from the LRT lookup is used as the new Request-URI for the outgoing request.

When a request is routed rather than re-targeted, the Request-URI is not changed, but one or more Route headers may be inserted into the outgoing request. Sometimes a request which already contains Route headers will be routed without adding additional Route headers.

When the Net-Net SBC routes requests and the original Request-URI was not the Net-Net SBC itself, the URI from the LRT /ENUM lookup is added as the top Route: header including the "lr" parameter. The Request-URI then remains unchanged.

Whether the Session Director re-targets or routes a request depends on the following:

- The target (Request-URI) of the received request
- The presence of Route headers
- Local Policy Attributes,
- Registration Cache matching.

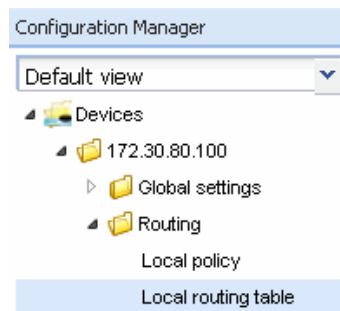
If the original target is the Net-Net SBC itself (i.e. the Request-URI contains the IP Address in the SIP interface the request was received on), the request is always retargeted. When the original target is not the Net-Net SBC and Local Policy is applied, the request will be re-targeted when the *policy attribute action* parameter is **replace-uri**. The request will also be re-targeted when the *policy attribute* specifies an ENUM or LRT lookup.

Retargeting requests can be configured in either the ENUM or LRT config depending on the request URI retrieval method chosen.

Instructions

To configure the Net-Net SBC to retarget or route request messages when performing an LRT lookup:

1. Expand the Configuration Manager slider > Routing.



2. Click Local routing table. The local routing table appears in the content area.
3. **Retarget requests**—Leave this parameter set to **enabled** for the Net-Net SBC to replace the Request-URI in the outgoing request. Select **disabled** from the drop-down list for the Net-Net SBC to route the request by looking to the Route header to determine where to send the message.
4. Click **Apply**.

Introduction

Net-Net SBC address translation is used to change a layer-5 endpoint name according to prescribed rules. Address translations can be performed on both the inbound and the outbound call legs independently, before and after routing occurs. Address translation is used for SIP, H.323, and SIP/H.323 interworking configurations.

Address translation takes place twice for both H.323 and SIP calls. The first address translation is applied to the incoming leg of the call, before the outgoing route is selected. The second address translation is applied to the outgoing leg of the call after the outgoing route is selected.

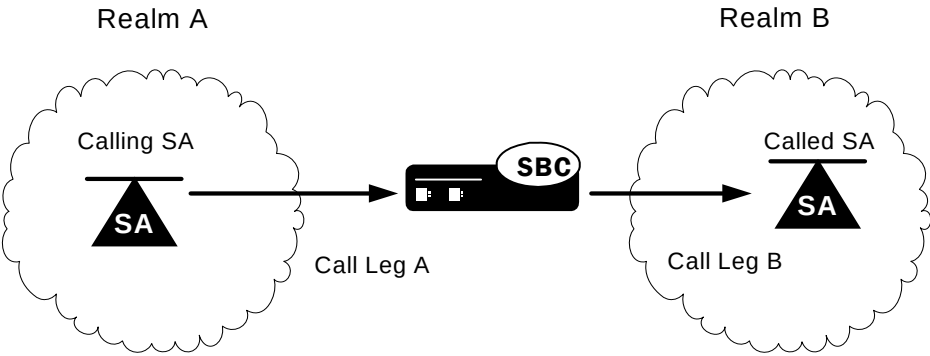
Address translation can be used to strip address prefixes added by external gateways. It can also be used to add a string tag to an address in order to implement a local policy routing scheme, and then remove the tag upon egress from the Net-Net SBC. The most common use of address translation is to add or remove a “1” or a “+” from a phone number sent from or addressed to a device.

Address Translation Implementation

Net-Net SBC address translations are implemented in three steps. First, you define the individual address translation rules. Next, the defined rules are grouped in a specified order, called translation profiles, to apply to calling and called numbers. Finally, translation profiles are attached to either session agents or realms in the session agent element or realm configuration element.

Address translations attached to session agents take precedence over address translations attached to realms. If no address translation is applied to a session agent, then the Net-Net SBC will use the address translation applied to a realm. If an address translation is applied to both a realm and session agent, the translation attached to the session agent will apply. If session agents and realms have no associated translations, then all numbers will remain in their original forms as they pass through the Net-Net SBC.

Within each realm or session agent, the address translation is applied to either the incoming or outgoing call leg. This distinction between incoming and outgoing calls is made from the point of view of the Net-Net SBC. The following diagram illustrates the address translation concept.



The following table shows you which parameters to apply an address translation ID in order to affect the corresponding leg of the call as shown in the illustration.

Leg	Calling SA	Called SA	Realm A	Realm B
A	IN Translation ID		IN Translation ID	
B		OUT Translation ID		OUT Translation ID

Address Translation in SIP URIs

Address translations only change the user portion of the URI. A typical SIP URI looks like `sip:user@hostname`. The user portion can take the form of either a phone number or any other string used for identification purposes.

Within the SIP header exists a Request URI, a To URI, and a From URI. The address translation element’s rules calling parameter modifies the From URI, while the rules called parameter modifies the Request URI and the To URI.

Address Translation in H.323 Messages

Because H.323 messages explicitly define the calling and called parties, the correspondence is exactly the same between the endpoints and configuration parameters. The H.323 calling party corresponds to the address translation element’s rules calling parameter. The H.323 called party corresponds to the address translation element’s rules called parameter.

Address Translation Configuration Overview

This section describes the procedure to create and apply address translations on the Net-Net SBC.

Configuring the address translation feature requires the following steps:

1. Create and configure individual translation rules.
2. Create translation profiles that group these rules for application to session agents and realms.
3. Apply these groups of rules on a per session agent or per realm basis using the appropriate fields in the session agent or realm configuration elements.

Creating Translation Rules

When you create a translation rule, you specify the type of translation to be performed, the addition or deletion to be made, and where in the address that change takes place.

First determine the type of translation to perform. The following table describes the three types of address translations.

Field Value	Description
add	Adds a character or string of characters to the address.
delete	Deletes a character or string of characters from the address.
replace	Replaces a character or string of characters within the address. Replace works by first applying the delete parameter then by applying the add parameter.

After you set the translation type, you define the string to add or delete. The wildcard term for a string to delete is the at-sign, "@". Finally, you specify the character position in the address to make the addition or deletion.

The character position where an add or delete occurs is called an index. The index starts at 0 (immediately before the leftmost character) and increases by 1 for every position to the right you move. If you want to add a string at the end of a number, enter a dollar-sign, "\$".

Creating Translation Profiles

A translation profile defines how translation rules are applied to calling and called numbers. A translation profile can contain multiple translation rules, it groups rules together and allows them to be referenced by one identifier.

When creating a translation profile, there are two fields that must be filled in, the "From" translation rules and the "To" translation rules. In the "From" translation rules enter the configured address translation that you want to affect the incoming traffic. In the "To" translation rules enter the configured address translation that you want to affect the outgoing traffic in this parameter.

The Net-Net SBC applies the translation rules in the order in which they are entered. They are applied cumulatively. For example, if this field is configured with a value of "rule1 rule2 rule3", rule1 will be applied to the original number first, rule2 second, and rule3 last.

Applying Translation Profiles

Translation profiles can be applied to both session agents and realms. Both session agents and realms contain the two parameters that denote inbound and outbound call legs—in translation ID and out translation ID. These two fields are populated with address translation element IDs.

If none of these fields are populated, no address translation will take place and the original address will remain unchanged as it traverses the Net-Net SBC. Further, any address translation applied to a session agent takes precedence over one applied to a realm.

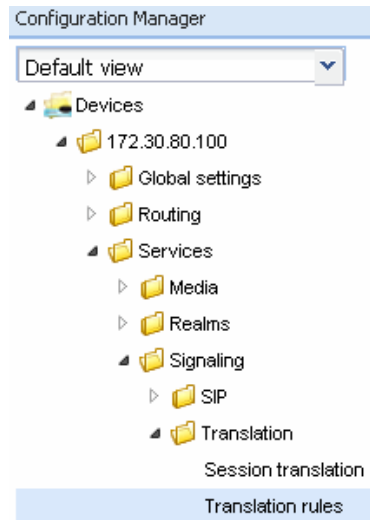
Configuring Address Translation Profiles and Rules

This section explains how to configure address translation. It also provides sample configurations for your reference.

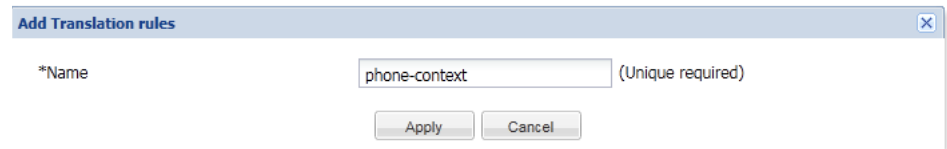
Configuring Translation Profiles and Rules

To create translation rules:

1. Expand the Configuration Manager slider > Services > Signaling > Translation.



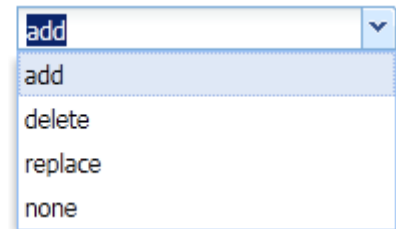
2. Click Translation rules.
3. Click **Add**. The Add Translation rules dialog box appears.
4. **Name**—Enter a name and a description for the translation rule.



5. Click **Apply**. The rule name appears in the translation rule table.
6. Click the rule name in the table and click **Edit**. The Edit Translation rules window appears.
7. **Description**—Edit the existing information if needed.
8. **Type**—Select the type of rule to create from the combination box.
 - **add**—Creates a translation rule that adds strings.
 - **delete**—Creates a translation rule that delete strings.

- **replace**—Creates a translation rule that replace one string with another.

Type



Adding a String

To create a rule that adds a string:

1. **String**—Enter the string you want to the rule to add.
2. **Index**—Enter the position where you want to add the string. Zero (0) is the left most position. If you want to add a string at the end of an number, enter a dollar-sign "\$".

Add

String

Index

Deleting a String

To create a rule that deletes a string:

1. **String**—Enter the text you want to delete in the String field.
2. **Index**—If you are deleting a portion of the address string, enter the index number in the Index text box. Remember to enter the number of characters you are deleting in the form of at-signs "@" in the delete string field.

The first matched string will be deleted, any remaining strings that match will remain. For example, if the address is 187521865 and the string to delete is "18," only the first instance of "18" will be deleted. The second instance will remain after translation.

Delete

String

string text here

Index

0

Replacing a String

To create a translation rule that replaces a string:

1. **Add String**—Enter the new string you want to use in the Add section text boxes.
2. **Delete String**—Enter the existing string you want to replace in the Delete section textboxes.

translationRules

*Name

phone-context

Type

replace

Add

String

new string to be used

Index

0

Delete

String

existing string to be replaced

Index

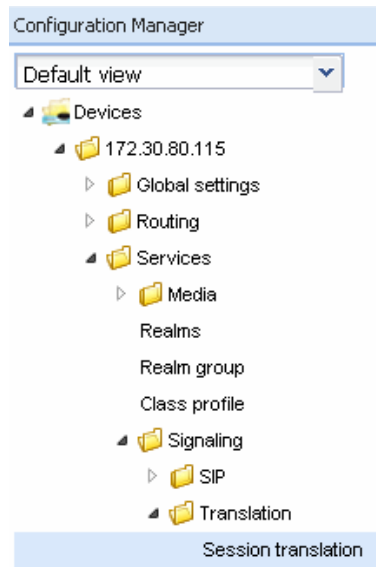
0

3. Click **Apply**.

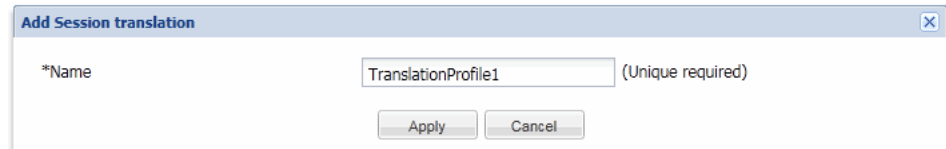
Translation Rules

To create a translation rule:

1. Expand the Configuration Manager slider > Services > Signaling > Translation.

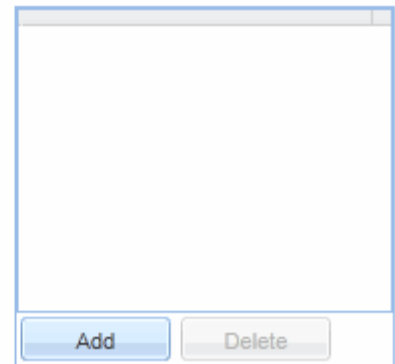


2. Click Session translation.
3. Click **Add**. The Add Session translation dialog box appears.
4. **Name**—Enter a name and description for the translation profile and click **Apply**. The new profile appears in the translation profile table.



5. Click the translation profile in the table and click **Edit**.
6. From translation rules—Click **Add** to choose the rules you want applied to incoming traffic.

From translation rules



The Add From translation rules dialog box appears.

7. **Rule calling**—Select the first translation rule to apply from the combination box.

8. Click **Apply** to continue.

9. Repeat this process until you have added all the Translation Rules to the From translation rules for this Translation profile. Multiple rules can be entered in this field. The order you enter them in is the order in which they are applied.

From translation rules

10. **“To” Translation rules**—Repeat steps 7-10 to create the translation profile applied to outgoing traffic. Multiple rules can be entered in this field. The order you enter them in is the order in which they are applied.

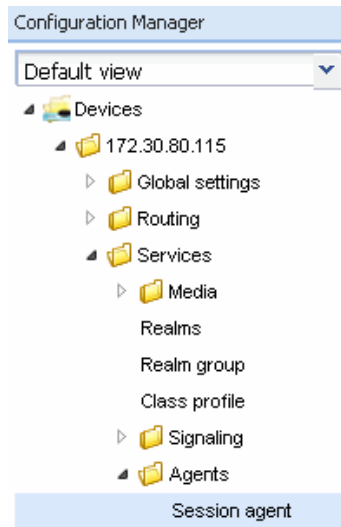
To translation rules

11. Click **Apply** at the bottom of the content area to complete creating translation rules.

Applying Profiles to a Session Agent

To configure address translation for a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a SIP session agent from the table and click **Edit**.
4. Click the Translation and manipulation tab at the top of the content area. Scroll to Address translation profiles.
5. **Inbound**—Select a translation profile for Inbound sessions from the combination box, affecting incoming traffic. The combination box contains all configured translation profiles. You can only select one profile for each category.
6. **Outbound**—Select a translation profile for Outbound sessions from the combination box, affecting outgoing traffic. The combination box contains all configured translation profiles. You can only select one profile for each category..

Address translation profiles

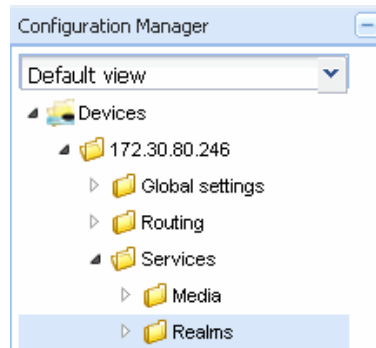
Inbound	
Outbound	

7. Click **Apply** at the bottom of the content area to complete applying the translation profile.

Applying Translation Profiles to a Realm

To configure address translation for a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm and click **Edit**.
4. Scroll to Address translation profiles.
5. **Inbound**—Select a translation profile for Inbound sessions from the combination box, affecting incoming traffic. The combination box contains all configured translation profiles. You can only select one profile for each category.
6. **Outbound**—Select a translation profile for Outbound sessions from the combination box, affecting outgoing traffic. The combination box contains all configured translation profiles. You can only select one profile for each category..

Address translation profiles

Inbound	
Outbound	

7. Click **Apply**.

Other Translations

SIP NAT Translations

There are other translations that occur by way of SIP NAT functionality acting on the SIP R-URI, From-URI, and To URI headers. The translation of URIs in the SIP message occurs as messages are received and sent from the Net-Net system's SIP proxy. These translations create a bridge between the external and home realms and remove all references to the original IPv4 addressing from the packets sent to the destination network.

The purpose of this translation is to prevent private IPv4 addresses from appearing in SIP message URIs while traveling through the public network. This aspect of the SIP NAT's functionality involves either translating the private address to a public address or encrypting the private address into the URI.

For information about configuring these additional number mappings, see the *SIP Signaling Service* chapter of this guide.

FQDN Mapping

The Net-Net SBC maps FQDNs that appear in certain headers of incoming SIP messages to the IPv4 address that the Net-Net SBC inserts in outgoing SIP contact headers. The mapped FQDNs are restored in the SIP headers in messages that are sent back to the originator.

This feature is useful to carriers that use IPv4 addresses in the SIP From address to create trunk groups in a PSX for routing purposes. When the carrier's peer uses FQDNs, the carrier is forced to create trunk groups for each possible FQDN that it might receive from a given peer. Similarly, this can apply to SIP Contact and P-asserted-identity headers.

For information about configuring these additional number mappings, see the *SIP Signaling Service* chapter of this guide.

Overview

This chapter describes how to configure the Net-Net SBC for call admission control and Quality of Service (QoS) monitoring. Call admission control lets you manage call traffic based on several different policies. It is aimed at managing call admission rates in the network, enabling you to maintain suitable QoS levels. A new call is admitted only if it meets the requirements.

QoS reporting provides you with real-time evaluation of network and route performance. It lets you contrast internal domain and external domain performance and facilitates SLA verification and traffic engineering.

About Call Admission Control

The Net-Net SBC provides call admission control capabilities based on the following policies:

- Bandwidth (single and multi-level policies)
- Session capacity
- Session rate (sustained and burst)

Note: In order to provide admission control for networks to which the Net-Net system is not directly connected, you need to define multiple realms per network interface.

Bandwidth-Based Admission Control

The Net-Net SBC is a policy enforcement point for bandwidth-based call admission control. Sessions are admitted or rejected based on bandwidth policies, configured on the Net-Net SBC for each realm.

To manage bandwidth consumption of a network's overall capacity, you can configure aggregate bandwidth policies for each realm. See *Configuring Realms* for additional information.

As the Net-Net SBC processes call requests to and from a particular realm, the bandwidth consumed for the call is decremented from the bandwidth pool for that realm. The Net-Net SBC determines the required bandwidth from the SDP/H.245 information for SIP and from the OLC sent in the SETUP message for H.323. Any request that would cause the bandwidth constraint to be exceeded is rejected with a SIP 503 Service Unavailable or an H.323 Release Complete.

For example, if an incoming SIP message requests PCMU for a payload/encoding name, a zero (0) payload type, and an 8000 cycle clock rate, the Net-Net SBC must determine how much bandwidth is needed.

To accomplish this task, the system checks the media profile values and reserves the bandwidth required for flows. If the required bandwidth for the new flow exceeds the available bandwidth at the time of the request, the Net-Net system rejects the session.

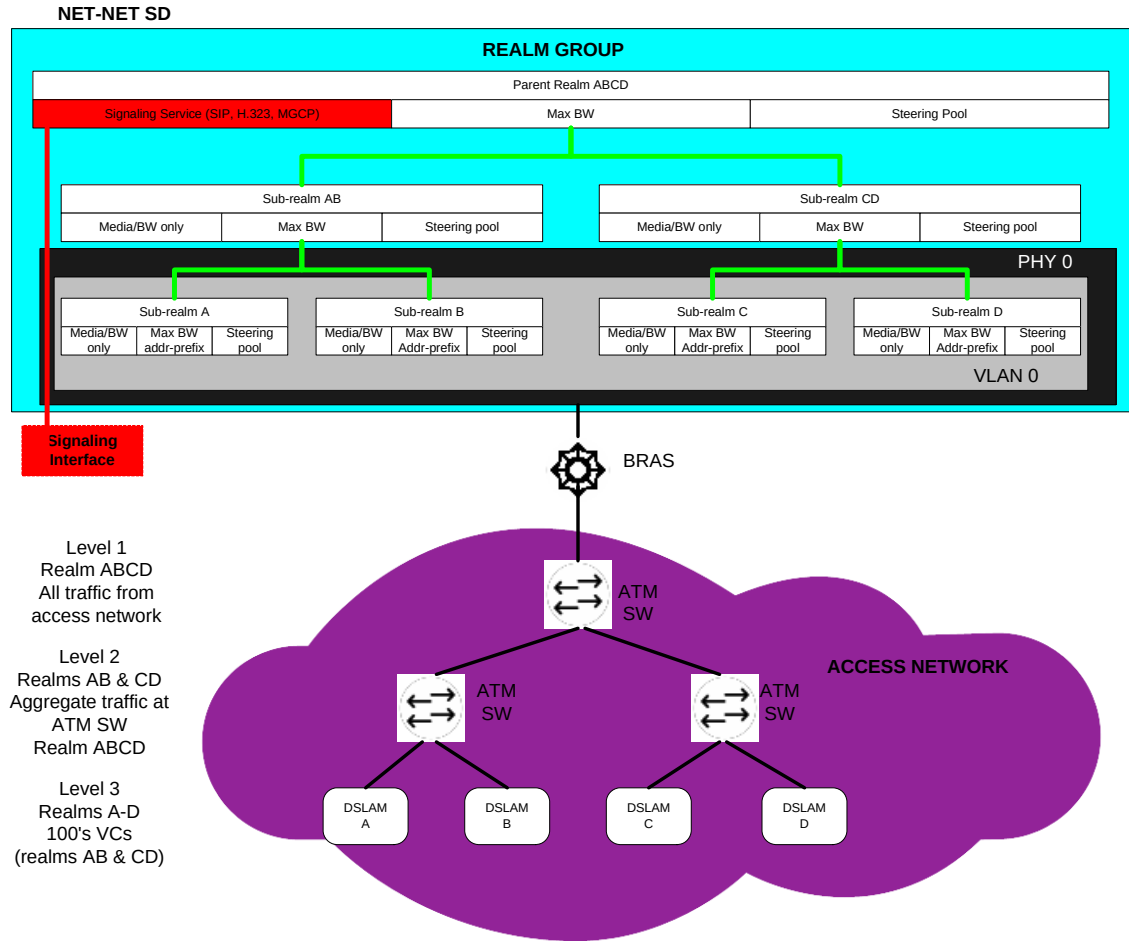
With these mechanisms, the Net-Net system provides bandwidth-based admission control.

Multi-Level Bandwidth Policy Nesting

Multi-level nesting of bandwidth policy enforcement addresses the following issues:

- Bandwidth over-subscription: access or transit transport networks are aggregated and/or oversubscribed. For example, digital subscriber lines (DSL), Frame Relay (FR), and Asynchronous Transfer Mode (ATM). Admission control policies must reflect access network topology.
- Bandwidth partitioning for multiple services: access or transit bandwidth is partitioned among multiple service profiles (for example, SIP and MGCP) in the same customer network.
- Multi-site VPN environments: admission control must be applied at the site level as well as the VPN level.

The following example illustrates different scenarios; in each there are two or more levels of admission control required. Nested admission control is best depicted by the DSL broadband example.



In DSL access networks, ATM network bandwidth is typically oversubscribed at rates up to 400/1. At Level 3 (above), hundreds of users virtual circuits (VCs) are aggregated to a smaller set of virtual paths (VPs) at each DSLAM. At Level 2, many virtual paths are aggregated at the first ATM switch. Finally, at Level 1, all traffic from all subscribers in the access network is aggregated at the BRAS. Each level of aggregation is oversubscribed, creating the need to perform admission control at each level.

From a Net-Net SBC perspective, multiple tiers of realms are supported, each with its unique bandwidth policy. Only the lowest order realm (Level 3) requires an address prefix (that assigned to the DSLAM) that must be used by the Net-Net SBC to determine in which realm a user resides. When a call request to or from a particular user is received, the Net-Net SBC checks each realm in the path to determine whether sufficient bandwidth is available to place the call.

Session Capacity- and Rate-based Admission Control

A session agent defines a signaling endpoint. It is a next hop signaling entity that can be configured to apply traffic shaping attributes. You can define concurrent session capacity and rate attributes for each session agent.

You can configure a set of attributes and constraints for each session agent to support session access control. In this configuration, the Net-Net SBC only accepts requests from configured session agents. And you can set up session admission control so that the Net-Net SBC limits the number of concurrent inbound and outbound sessions for any known service element.

The Net-Net SBC denies a call request to any destination that has exceeded its configured policies for session capacity and session rate. The Net-Net SBC might reject the call request back to the originator. If multiple destinations are available, the Net-Net SBC will check current capacity and rate for each destination and attempt to route the call only to destinations whose policy limits have not been reached.

You assign a media profile to a session agent and indicate whether the transport protocol is SIP or H.323. If the protocol is H.323, you need to indicate whether the session agent is a gateway or a gatekeeper.

Asynchronous SIP-Diameter Communication

The Net-Net SBC's Diameter-based external policy server support offers an asynchronous mode which can address the following two situations:

One of the fundamental behaviors in the Net-Net SBC's Diameter model relies on the external policy server making an authorization decision which is then communicated to the Net-Net SBC. Part of the call authorization sequence of events involves the Net-Net SBC waiting for an external policy server's response before the Net-Net SBC can completely set up, modify, or update the call. The long pause that the endpoints experience, while the Net-Net SBC holds up SIP flows waiting for the external policy server's response, may lead to unnecessary call failure situations.

In some Diameter-based external policy server deployments, the media traverses a CMTS at the edge of the network; the CMTS gates may be established by a policy server to dynamically enable QoS to/from a UE toward another UE. If no gate is established then the media should still traverse the CMTS and be admitted to the network with a "best-effort" network path.

Synchronous Model

The Net-Net SBC's default external policy server model may be referred to as the synchronous model; The Net-Net SBC sends a Policy Server request based on a

SIP/SDP trigger point, the SIP signaling is held until a response is returned from the policy sever. For example:



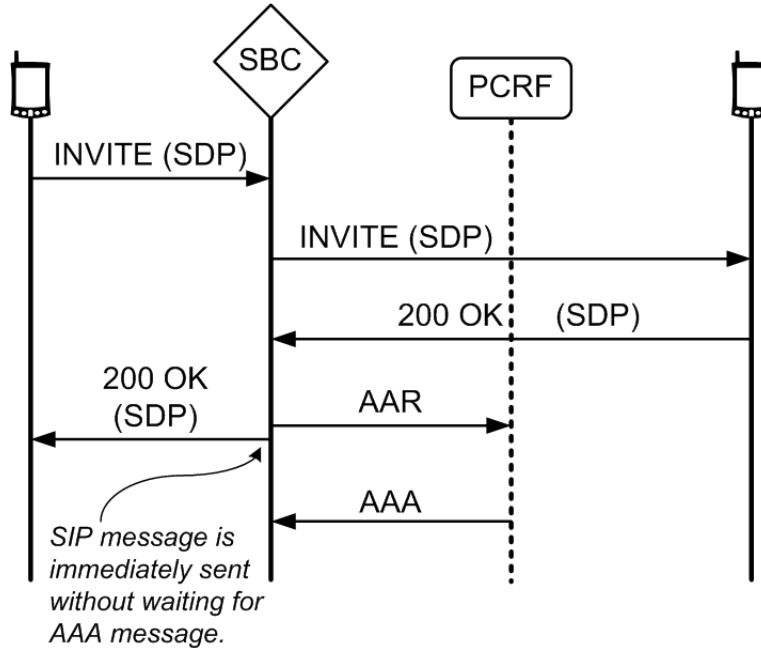
Note: In the above diagram, the **reserve incomplete** parameter has been set to **disabled**.

Asynchronous Model

In the asynchronous model the request that the Net-Net SBC's sends to the Policy Server flows in an asynchronous state with respect to SIP messaging. Thus the Net-Net SBC allows the SIP session to proceed naturally, and does not pause for outstanding Policy Server answers to be received.

It is crucial to understand that the Policy Server's ultimate success or failure answer, or timeout, as related to a SIP session does not impact the establishment of the SIP session itself.

The following diagram represents the asynchronous model where SIP sessions always succeed. SIP signaling is never delayed and as a result, call set-up time is decreased.



Note: In the above diagram, the Early resource reservation parameter has been set to **disabled**.

Caution

Acme Packet recommends that for each Net-Net SBC, the same model (Synchronous / Asynchronous) is used for all external policy servers configuration instances. Failure to follow this guideline may result in complex interactions from a timing perspective which may lead to a negative end user experience.

Serialized Diameter Messaging

Access networks with complex policy server structures might introduce the possibility of un-ordering Diameter requests into a given CMTS, even if Diameter message delivery was correctly ordered on the TCP connection between the Net-Net SBC and a lower-tiered external policy server.

The Net-Net SBC can prevent the external policy server from receiving out-of-order messages at the application layer by serializing them. The Net-Net SBC serializes Diameter messages by ensuring a Diameter request for one sessionID is not sent until an answer is received from the previous request for the same sessionID. The Net-Net SBC applies this constraint while waiting for Diameter response or when considering a Diameter request timeout (15 seconds).

Serialized Diameter messaging is always enforced for Diameter-based External Policy Server communication.

High Availability Considerations

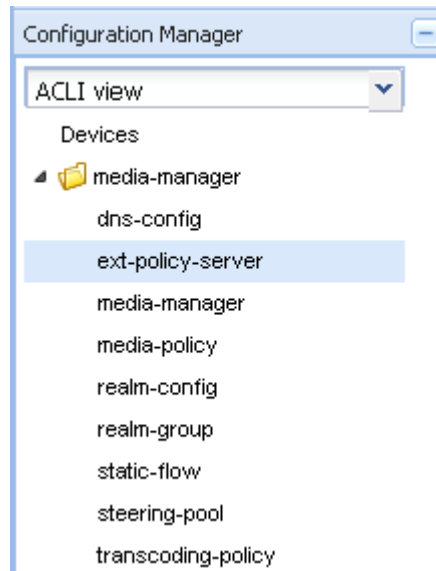
Upon a failover, the Net-Net SBC may not guarantee the synchronicity of Diameter messages for active sessions. For example, if the active Net-Net SBC sent an AAR and an AAA is yet to be received, upon failover the newly active SBC may send a 2nd AAR or an STR without receipt of the initial AAA. This creates an unavoidable race condition. Such rare cases must be handled appropriately within the Quality of Service subsystem (Policy Servers and CMTSs).

Configuring the External Bandwidth Manager

You must configure media profiles to utilize bandwidth policing. This feature can only be enabled in the ACLI view.

To configure the external bandwidth manager to run in asynchronous mode:

1. Expand the Configuration Manager slider.



2. Select ACLI view from the combination box.
3. Expand the media-manager folder and click ext-policy-server.
4. **asynchronous-mode**—Select enabled to enable this external policy server to run in asynchronous mode.
 asynchronous-mode enabled
5. Click **Apply**.

Bandwidth CAC Fallback Based on ICMP Failure

For networks where backup links (operating in active-standby mode) from CE-routers to the MPLS backbone are provisioned with less bandwidth than the primary links, the Net-Net SBC can:

- Detect remote link failures
- Trigger bandwidth updates at the realm level when using backup links
- Detect remote link fallback to primary

To do so, the Net-Net SBC monitors the primary link status using ICMP echo requests (or pings). It issues the pings at regular intervals, forming a heartbeat

mechanism. The CE-router can respond to these pings on the primary link, which is represented by the WAN IP address. When this link fails over, the backup link assumes the same WAN IP address but is not responsive to the pings. This way, the Net-Net SBC determines failover when the ICMP ping fails.

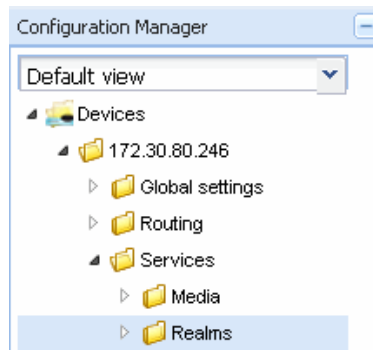
When there is an ICMP ping failure, the Net-Net SBC adjusts the realm's available bandwidth pool from its maximum bandwidth setting to its fallback setting. If the fallback amount is less than the maximum amount, it is possible for the Net-Net SBC to start rejecting calls. It does so until enough calls are released to free adequate bandwidth to stay under the fallback limit and still accept calls.

Enabling Bandwidth CAC Fallback

You can set up ICMP heartbeats and fallback bandwidth pools in the realm configuration. Leaving the **detection multiplier**, **advertisement interval**, or **target IP address** parameters blank or set to zero turns the feature off.

To enable bandwidth CAC fallback based on ICMP failure:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the realm table and click **Edit**. The Edit Realm window appears.
4. Scroll to CAC fallback on ICMP failure.
5. **Bandwidth**—Enter the amount of bandwidth you want available once the Net-Net SBC has determined that the target is unreachable.

If the fallback amount is less than the **bandwidth limit** value, the Net-Net SBC might start to reject calls. It does so until enough calls are released to free adequate bandwidth to stay under the fallback limit and still accept calls

6. **Detection multiplier (secs)**—Enter the multiplier in seconds you want to use when determining how long to send ICMP pings before considering a target unreachable. This number multiplied by the time you set for the **advertisement interval** determines the length of time. For example, if you set this parameter to 10 and the advertisement interval to 20, the Net-Net SBC will send ICMP pings for 120 seconds before declaring the target unreachable.
7. **Advertisement interval (secs)**—Enter the time in seconds between ICMP pings the Net-Net SBC sends to the target. The default is 0.
8. **Target IP address**—Enter the IP address to which the Net-Net SBC should send the ICMP pings so that it can detect when they fail and it needs to switch to the fallback bandwidth for the realm. The default is 0.

CAC fallback on ICMP failure

Bandwidth (kilobits per second)	50
Detection multiplier (sec)	10
Advertisement interval (sec)	20
Target IP address	172.30.10.15

9. Click **Apply**.

When it determines a target is unreachable, the Net-Net SBC issues the `apSysMgmtRealmICMPReachableTrap` indicating ICMP heartbeat failure and sends the corresponding alarm. Because it can be sent for different realms, the trap might go out multiple times. However, the alarm appears only once in the alarm display. It will be cleared with the `apSysMgmtRealmICMPReachableClearTrap` only when all instances are cleared. See the *MIB Reference Guide* for more information.

Bandwidth CAC for Aggregate Emergency Sessions

You can configure the maximum amount of bandwidth on your Net-Net SBC you want used specifically for priority (emergency) calls in the realm configuration's **Max priority bandwidth** parameter. You set this limit on a per-realm basis, and the limit is enforced for nested realms. Setting a bandwidth limit specifically for priority calls allows the Net-Net SBC to reject calls exceeding the threshold, and also to accept calls that exceed the bandwidth limit for non-priority calls.

The bandwidth limit for emergency calls operates in conjunction with the bandwidth limits you can set for all other types of calls. When an emergency call comes in, the Net-Net SBC checks the non-priority bandwidth limit. If bandwidth is sufficient, the call goes through and the Net-Net SBC decrements the bandwidth used from the pool of the amount available.

However, if a priority call exceeds the **Bandwidth limit** setting, the Net-Net SBC checks the **Max priority bandwidth** parameter. If it is within the limit for priority calls, the system allows the call and decrements the amount of used bandwidth from what is available.

When there is not enough bandwidth in either the priority or non-priority pool, the Net-Net SBC rejects the call with the corresponding error code and reason phrase.

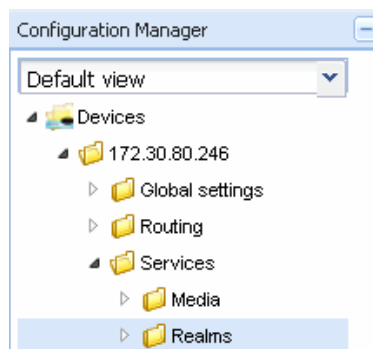
Any bandwidth subtracted from either pool during a session is returned to that pool as soon as the session ends.

Configuring Bandwidth CAC for Aggregate Emergency Sessions

You configure bandwidth CAC for priority calls on a per-realm basis. Note that this parameter honors the hierarchy of nested realms if you have them configured.

To enable bandwidth CAC for aggregate emergency sessions:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure in the realm table and click **Edit**.
4. Click the Access control tab.
5. **Max priority bandwidth (bits per second)**—Enter the amount of bandwidth you want to use for priority (emergency) calls in bits per second. Note that the system first checks the **Bandwidth limit** parameter, and allows the call if the value you set for priority calls is sufficient. If there is not enough priority and non-priority bandwidth allotted for an incoming call, the Net-Net SBC rejects it.

This parameter defaults to 0. You can enter any value between 0 and 999999999.

Maximum priority bandwidth (bits per second)

6. Click **Apply**.

Configuring Admission Control

This section explains how to session agents for admission control.

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by `admin@console`

Last modified date `2011-04-26 18:27:37`

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

`EMS_admin_172.30.80.2.UTC`

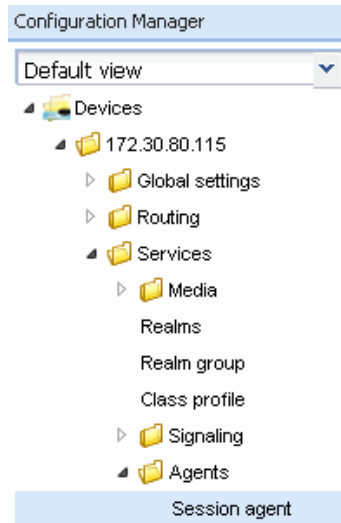
The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Adding a Session Agent

This section describes how to add, configure, and delete admission control for your Net-Net SBC.

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click **Add**. The Add Session agent box appears.
4. **Name (FQDN)**—Enter the name you want to use for this session agent in FQDN format.

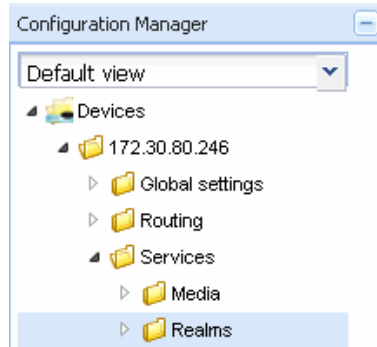
The 'Add Session agent' dialog box is shown. It has a text input field for '*Name(FQDN)' containing '172.114.10.30' with a '(Unique required)' label. Below the field are 'Apply' and 'Cancel' buttons.

5. Select the session agent from the table and click **Edit**.
6. **IP Address (required for SIP pings to test SA availability)**—Enter an IP address for SIP pings to test session agent availability.
7. **Protocol**—Select the protocol from the combination box to be used for this session agent.
8. Click **OK**. The session agent appears in the table in the session agent window.

The screenshot shows the configuration fields for a session agent. The 'IP Address (required for SIP pings to test SA availability)' field contains '172.10.30.25'. The 'Protocol' dropdown menu is open, showing 'SIP' selected, with 'H323' and 'SIP' as options.

Configuring a Session Agent

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the session agent you want to configure and click **Edit**.
4. Click the Constraints tab at the top of the content area.

Configuring Session Constraints

1. Click the Constraints tab at the top of the content area.
2. **Session constraints**—Select enabled from the combination box to activate SIP session agent constraints.
3. **Max. number of sessions (#)**—Enter the maximum number of inbound and outbound sessions.
 - minimum: zero (0) is the default value and means there is no limit
 - maximum: $2^{32} - 1$

4. **Max. INVITE burst rate (# sessions per sec)**—Enter the maximum SIP INVITE burst rate in sessions per second. This is the number of session invitations allowed to be sent to or received from the session agent within the configured burst rate window value.

SIP session invitations arrive at and leave from the session agent in intermittent bursts. By entering a value in this field, you can limit the amount of session invitations that are allowed to arrive at and leave from the session agent.

For example, if you enter a value of 50 here and a value of 60 (seconds) for the burst rate window constraint, no more than 50 session invitations can arrive at or leave from the session agent in that 60 second time frame (window). Within that 60-second window, any sessions over the limit of 50 are rejected.

- minimum: zero (0) session invitations per second
 - maximum: $2^{32} - 1$ session invitations per second
5. **Max. INVITE sustained rate (# sessions per sec)**—Enter the maximum SIP INVITE sustained rate in sessions per second. This is the maximum rate of session invitations allowed to be sent to or received from the session agent within the current window. The value you set here must be larger than the value you set for the maximum burst rate constraint.
 - minimum: zero (0) invitations per seconds
 - maximum: $2^{32} - 1$ invitations per seconds

The current rate is determined by counting the number of session invitations processed within a configured time period and dividing that number by the time period. By entering a value in this field, you can limit the amount of session

invitations that are allowed to arrive at and leave from the session agent over a sustained period of time.

For the sustained rate, the Net-Net system maintains a current and previous window size. The period of time over which the rate is calculated is always between one and two window sizes.

6. **Max. inbound sessions (#)**—Enter the maximum number of simultaneous inbound sessions.

- minimum: zero (0) is the default value and means there is no limit
- maximum: $2^{32} - 1$

The value you enter here cannot be larger than the maximum sessions value.

7. **Max. outbound sessions (#)**—Enter the maximum number of simultaneous outbound sessions.

- minimum: zero (0) is the default value and means there is no limit
- maximum: $2^{32} - 1$

The value you enter here cannot be larger than the maximum sessions value.

8. **Burst rate window (sec)**—Enter a new burst rate window in seconds. The term *window* refers to the period of time over which the burst rate is computed. Refer to the maximum burst rate information.

The value you set here must be smaller than the value you set for the maximum burst rate constraint.

For example, if you enter a value of 5000 here and a value of 3600 (seconds) for the sustain rate window constraint, no more than 5000 session invitations can arrive at or leave from the session agent in any given 3600 second time frame (window). Within that 3600-second window, sessions over the 5000 limit are rejected.

9. **Sustained rate window (sec)**—Enter a new sustained rate window in seconds. The value you set here must be larger than the value you set for the maximum sustain rate constraint.

- minimum: zero (0) seconds
- maximum: $2^{32} - 1$ seconds

10. **Max. rate of registration messages sent to SA (reg/sec)**—Enter the time in seconds after which the SIP proxy should resume sending session invitations to the session agent. This value only takes effect when the SIP proxy stops sending invitations because a constraint is exceeded.

- minimum: zero (0) seconds

- maximum: $2^{32} - 1$ seconds

Session constraints	enabled ▼
Max. number of sessions (#)	100
Max. INVITE burst rate (# sessions per sec)	
Max. INVITE sustained rate (# sessions per sec)	0
Max. inbound sessions (#)	0
Max. inbound burst rate (# sessions per sec)	0
Max. inbound sustain rate (# sessions per sec)	0
Max. outbound sessions (#)	0
Max. outbound burst rate (# sessions per sec)	0
Max. outbound sustain rate (# sessions per sec)	0
Burst rate window (sec)	0
Sustained rate window (sec)	0
Max. rate of registration messages sent to SA (reg/sec)	0

Configuring Timers

1. **After threshold reached before resuming INVITES to SA**—Enter the time in seconds the SIP proxy must wait between the time an invitation is sent to an SA and the time it tries again.
2. **After SA unresponsive before resuming INVITES to SA**—Enter the maximum number of registration messages per second you want sent to this session agent.
3. **After communication established before SA considered 'in service'**—Enter the time in seconds the session agent must be operational before it is considered in-service.

Time-to-wait timers (sec)

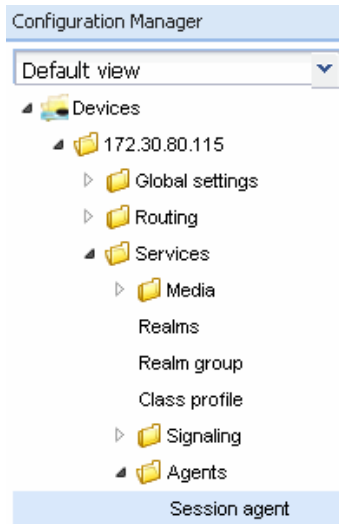
After threshold reached before resuming INVITES to SA	100
After SA unresponsive before resuming INVITES to SA	0
After communication established before SA considered 'in service'	30

4. Click **Apply**.

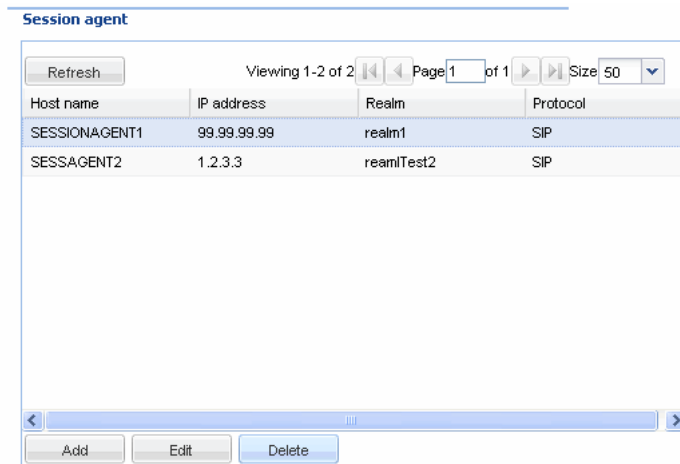
Deleting a Session Agent

To delete an existing session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select a session agent in the table you want to delete.

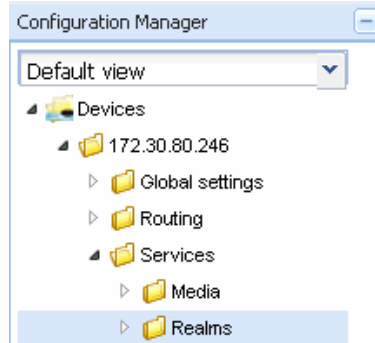


4. Click **Delete** at the bottom of the window. A confirmation window appears.
5. Click **Yes**. The session agent is deleted.

Configuring Realm Bandwidth

This section describes how to configure bandwidth control per realm for your Net-Net SBC.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure and click **Edit**.
4. **Bandwidth limit (kilobits per second)**—Set the maximum bandwidth resources available for this realm; the default is 0.

Realm services profiles

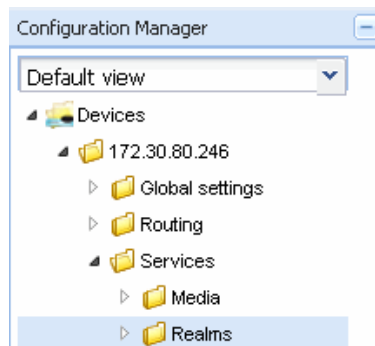
*Bandwidth limit(kilobits per second),Session admission control

5. Click **Apply** to complete configuring admission control for this realm.

SIP Admission Control

This section describes how to configure admission control based on an endpoint's registration state.

1. Expand the Configuration Manager slider > Services.



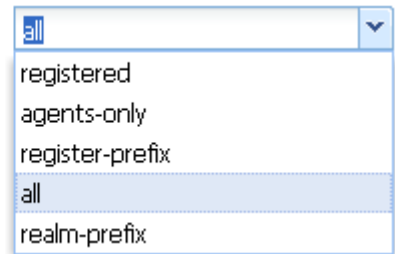
2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**. The Edit SIP interface window appears.
6. Scroll to the SIP Signaling Interface table, select an interface and click **Edit**.
7. **Anonymous mode**—Choose the criteria for admission control that you want applied when SIP requests are processed.

The following table lists the available options:

Option	Description
all	No ACL is applied and the all anonymous connections are allowed.
agents only	Only requests from configured session agents are processed. The Net-Net SBC responds to all other requests with a <i>forbidden</i> response.
realm prefix	Only requests from session agents and addresses matching the realm's address prefix are processed. All other requests are rejected with a 403 (Forbidden) response.
registered	Only requests from session agents and registered endpoints are processed. REGISTER allowed from any endpoint.
registered prefix	Only requests from session agent and registered endpoint addresses that match the realm's realm prefix are processed.

8. Click **Apply**. You will be returned to the SIP service window.

Anonymous mode

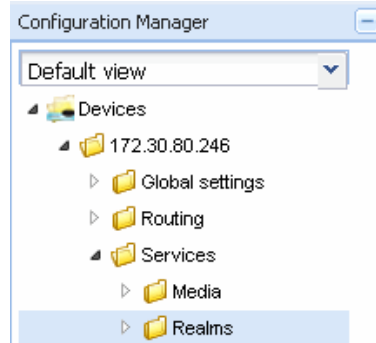


9. Click **Apply**.
10. Click **Apply**.

H.323 Admission Control

This section describes how to configure H.323 admission control on your Net-Net SBC.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select the H.323 Service/Protocol name you want to configure and click **Edit**.
6. Click the H.323 Service mode tab at the top of the content area.
7. **Service mode**—Select enabled from the combination box.
8. **Allowed connections**—Select the endpoints with source IP address within this realm's prefix checkbox to enable it.

The following table lists the available options.:

Option	Description
all	No ACL is applied and the all anonymous connections are allowed.
agents only	Only requests from configured session agents are processed.
realm prefix	Only requests from session agents and addresses matching the realm's address prefix are processed. All other requests are rejected.

Service mode

enabled

Allowed connections

all
agents-only
realm-prefix
all

9. Click **Apply**.

Conditional Bandwidth CAC for Media Release

The Net-Net SBC supports conditional call admission control (CAC) using the SIP profile configuration. You can allow the conditional admission of SIP calls that could potentially have their media released, instead of risking the possible rejection of those calls due to internal bandwidth limits.

About Conditional Bandwidth CAC for Media Release

The Net-Net SBC performs bandwidth CAC for SIP per realm, for each Address of Record (AoR) or IP address. The system checks bandwidth limits based on the codecs listed in SDP. If a new SIP INVITE contains codecs in an SDP message that exceed bandwidth available for a given resource, the system rejects that INVITE. This check occurs both on the ingress and egress sides of a call, and both sides must have enough available resources to support the call for it to be admitted.

In the case of calls where media is released, the Net-Net SBC does not count bandwidth consumed by the call. However, this exemption is not given until the media is actually released—and media release conditions are unknown at the time SIP INVITE is admitted. This is because an INVITE received on one side of the Net-Net SBC is only media-released when that INVITE is routed back through the Net-Net SBC as a hairpin or other multi-system media release. So there has to be enough bandwidth for the initial INVITE; otherwise, and even if the INVITE is a candidate for media release, it will be rejected.

When there is a significant volume of such calls—ones that are candidates for media release, but cannot be admitted because of CAC limits—it becomes important to admit them so long as they truly end in media release. This feature thus allows admission of SIP calls that might otherwise be rejected because of bandwidth limitations when the far-end of the call causes media to be released.

Details and Conditions

This feature applies in a two system scenario. In order to track a call as a candidate for provisional media release, the access-side Net-Net SBC adds a Require: header with an option tag to the INVITE or UPDATE message on egress. The option tag is configurable in the SIP configuration. The default is *com.acmepacket.cac*.

The following sections describe when the SIP INVITE or SIP UPDATE are:

- initially received by the Net-Net SBC
- received by the second Net-Net SBC

INVITEs/UPDATEs Initially Received By Net-Net SBC

When the Net-Net SBC first receives an INVITE or UPDATE message, it considers if it should be admitted provisionally or rejected outright due to CAC bandwidth constraints. If the INVITE or UPDATE is admitted provisionally, a Require: header is inserted on egress from the system.

The Net-Net SBC inserts the Require header on egress under these conditions:

- It receives an INVITE / UPDATE with no or a non-matching Require header.
- The **Egress conditional CAC admit state** parameter in the SIP profile on the egress realm, SIP interface, session agent is set to enabled in the egress realm
- The request would otherwise be rejected because of current bandwidth CAC limits in the ingress OR egress realms
- The call is a candidate for media release in the ingress realm

A call is considered a candidate for media release when the ingress realm has any of these parameters set to disabled:

- **Within this realm serviced by this Net-Net system only**
- **Within different realms in the same subnet serviced by this Net-Net system only**
- **Media released behind same IP**
- **Release media across network interfaces**

INVITEs/UPDATES Received by Second Net-Net SBC

The second Net-Net SBC receives the INVITE or UPDATE with the newly inserted Require: header. Standard SIP convention indicates that if the UAS receiving the request does not know how to handle the Require header, the request should be rejected.

When the following three conditions are met, the INVITE is permitted into the system for processing:

- The **Ingress conditional CAC admit state** in the SIP profile on the ingress realm, SIP interface, session agent parameter is set to enabled
- The **con-cac-tag** sip config option is configured to the same value as the received Require header's option tag
- The call is a candidate for media-release

The call is considered a candidate for media-release on the second system (indicated by the **Ingress conditional CAC admit state** parameter is set to enabled) when either the ingress or egress realms have any of these parameters set to disabled:

- **Within this realm serviced by this Net-Net system only**
- **Within different realms in the same subnet serviced by this Net-Net system only**
- **Media released behind same IP**
- **Release media across network interfaces**

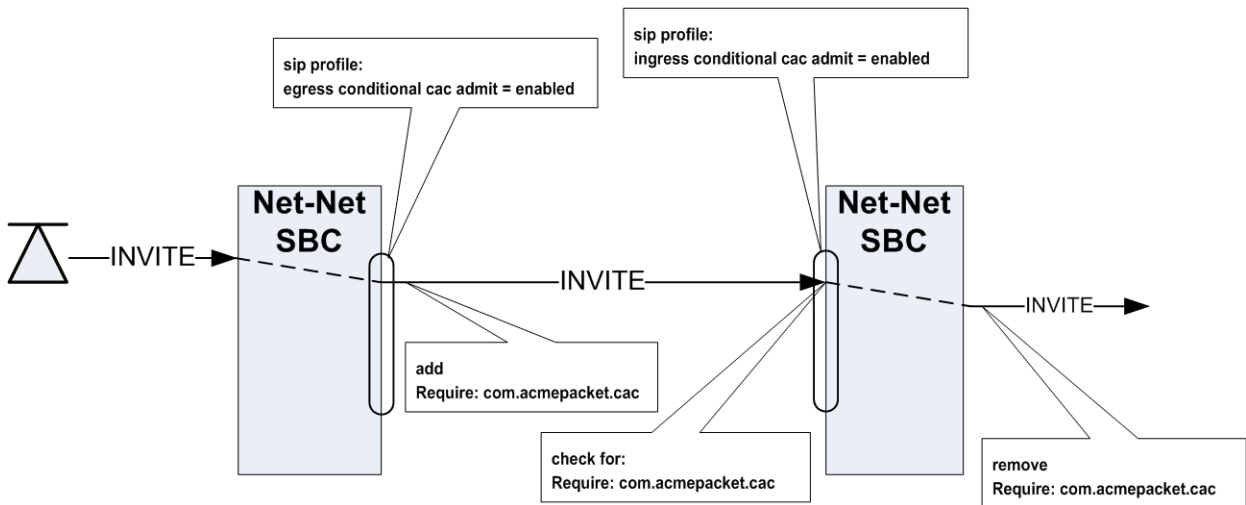
and the following parameter is set to enabled:

- **Within this realm serviced by different Net-Net systems (SIP only)**

If the call, as received by the second system is not considered a candidate for release, the INVITE or UPDATE is failed with a 503 "Insufficient Bandwidth" message.

After the INVITE has been processed by the Net-Net SBC, the Require: header is removed upon egress from the system.

The following diagram shows the two-system scenario:



Conditional Admission with Per-user CAC

In the event that the per-user CAC feature is also being used, and per-user CAC bandwidth is exceeded, the Net-Net SBC also uses this option tag mechanism. However, if the per-user CAC implementation does count bandwidth regardless of media-release, then the Net-Net SBC will reject calls exceeding the per-user CAC limits when it receives them.

On the second system, when the per-user CAC feature is being used, the Net-Net SBC will perform the same option tag mechanism based on if the **Ingress conditional CAC admit state** parameter is enabled.

Configuring a SIP Profile

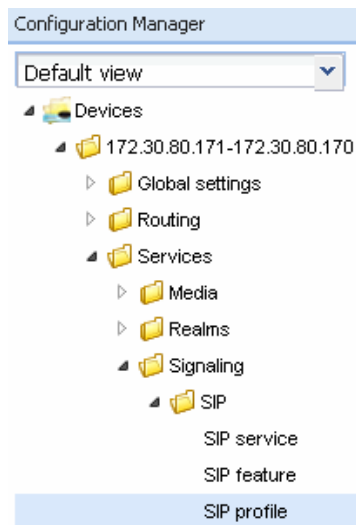
You enable this feature by first configuring a SIP profile, and then applying the profile to any of these:

- realm
- SIP interface
- SIP session agent

You can configure multiple SIP profiles. Though this configuration contains additional parameters, you do not have to use them for the conditional bandwidth CAC for media release.

To configure a SIP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.
3. Click **Add**. The Add SIP profile dialog box appears.
4. **Name**—Enter the name of the SIP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.

 A screenshot of the 'Add instance' dialog box. It has a title bar 'Add instance' with a close button. The main area is titled 'SIPPROFILE'. Below the title is a label '*Name' followed by a text input field containing 'sipProfile1' and the text '(Unique required)' to its right. At the bottom are two buttons: 'Apply' and 'Cancel'.

5. Click **Apply**. The new SIP profile appears in the SIP profile table.
6. Click the SIP profile in the table you want to configure and click **Edit**.
7. **Conditional CAC for media release on ingress**—Click **enabled** in the combination box to process an INVITE with a Require tag as received on an ingress interface. You can set this parameter to disabled if you do not want to use this feature on the ingress side. There is no default for this parameter.
8. **Conditional CAC for media release on egress**—Click **enabled** in the combination box if you want to use conditional bandwidth CAC for media release for calls that are first received by this system. This results in option tags being inserted on the INVITE's egress if the conditional CAC conditions are met.

You can set this parameter to disabled if you do not want to use this feature.
There is no default for this parameter.

CAC

Conditional CAC for media release on
ingress

enabled ▼

Conditional CAC for media release on
egress

enabled ▼
enabled
inherit
disabled

9. Click **Apply**.

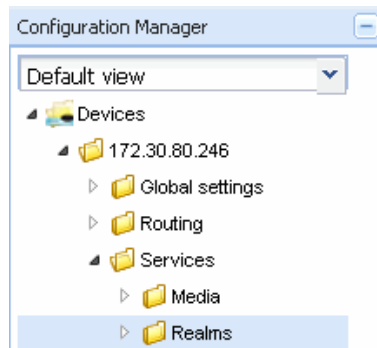
Applying a SIP Profile

Once you have configured one or more SIP profiles, you can apply them to realms, SIP interfaces, and SIP session agents.

Realms

To apply a SIP profile to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

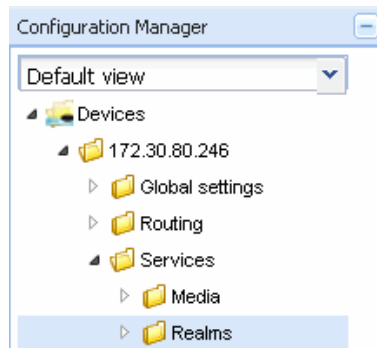
A dropdown menu with 'SIPprofile1' selected and displayed below the selection box.

4. Click **Apply**.

SIP Interface

To apply a SIP profile to a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**. The Edit Realm window appears.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.

6. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

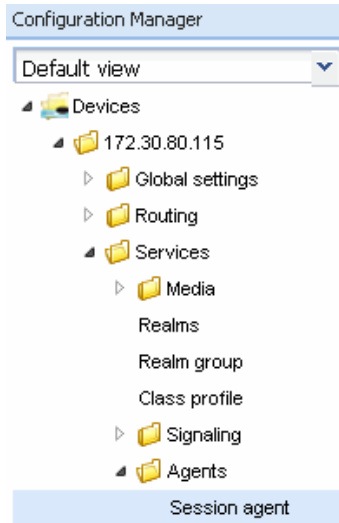
SIPprofile1

7. Click **Apply**.

Session Agent

To apply a SIP profile to a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

SIPprofile1

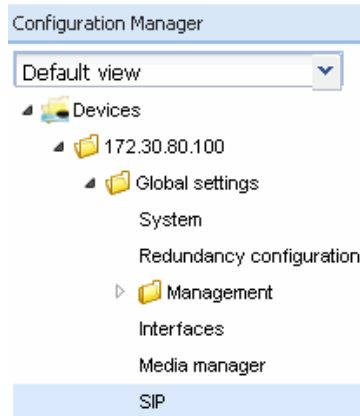
4. Click **Apply**.

Configuring Require Header Option Tag

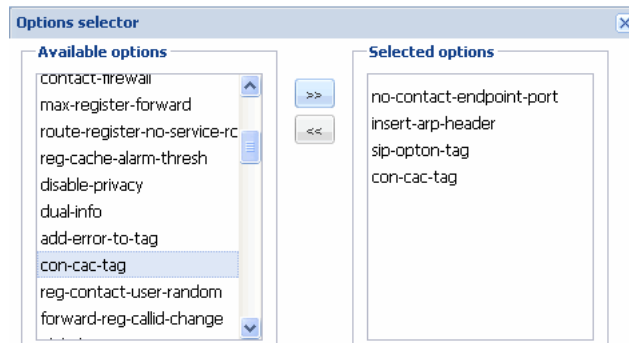
You may change the Require: header's option tag from the default `com.acmepacket.cac` to one of your own choosing. Remember that both systems' option tags must match exactly.

To configure the Require: header's option tag:

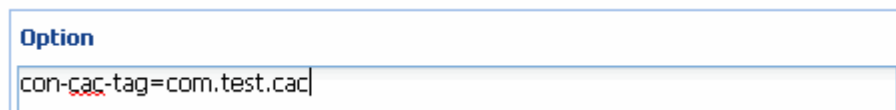
1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The SIP options window appears.
5. Click **con-cac-tag** in the Available options list to select it.



6. Click **>>** to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
7. **con-cac-tag**—Edit the Option text box to append this option with a new tag.



8. Click OK.
9. Click Apply.

Configuring CAC, Policing, and Marking for non-Audio, non-Video Media

You can enable the use of the AS modifier in the SDP and specify the percentage of headroom to be added.

Support for the AS Bandwidth Modifier

The following parameters provide support for AS bandwidth modifier:

- **SDP Bandwidth**—Enables or disables the use of the AS modifier in the SDP if the **Bandwidth requirements (Kbps)** and **SDP rate limit headroom (%)** EMS labels are not set to valid values in a corresponding media profile. The default value is **disabled**.
- **SDP rate limit headroom (%)**—Specifies the percentage of headroom to be added while using the AS bandwidth parameter, while calculating the **Maximum bandwidth allowed per flow (bytes/sec)** (rate limit for the RTP flow). The default value is zero (0). The valid range is:
 - Minimum—0
 - Maximum—100

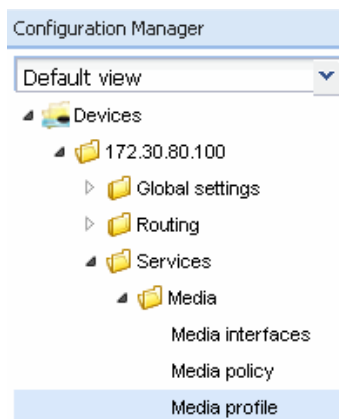
The following conditions apply to the use and application of these two parameters:

- If the amount of required bandwidth is not specified in the media profile (**Bandwidth requirements (Kbps)**) for the media type in the **m=** line of the SDP, then the value specified in the AS modifier is used. The Net-Net SBC only uses the AS value if you set the new **SDP Bandwidth** to **enabled**.
- If the average rate limit value for RTP flows is not specified in the media profile (**Maximum bandwidth allowed per flow (bytes/sec)**) for the media type in the **m=** line of the SDP, then the value specified in the AS modifier is used. The Net-Net SBC only uses the AS value if you set the new **SDP Bandwidth** to **enabled**. When calculating the average rate limit that it will use based on the AS modifier, the Net-Net SBC applies the percentage set in the **SDP rate limit headroom (%)** EMS label.
- The Net-Net SBC uses the value specified in the AS modifier (if **SDP Bandwidth** is enabled, and **Bandwidth requirements (Kbps)** is set to 0) along with the **Maximum bandwidth allowed** value set in the Realms window; this works the same way that the **Bandwidth requirements (Kbps)** parameter does.
- The Net-Net SBC uses the value specified in the AS modifier (if **SDP Bandwidth** is enabled, and **Bandwidth requirements (Kbps)** is set to 0) along with the **Bandwidth limit (kilobits per second)** value set in the Realms window; this works the same way that the **Bandwidth requirements (Kbps)** parameter does.

Enabling AS Modifier Support and Headroom

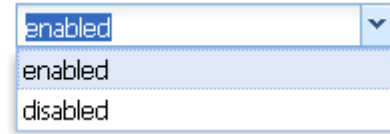
To enable AS modifier use and establish the percentage of headroom to use:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Select a media profile name in the table and click **Edit**. The Edit Media profile window appears.
4. **SDP Bandwidth**—Click the checkbox to enable this parameter to use the AS bandwidth modifier in the SDP. The default is disabled.

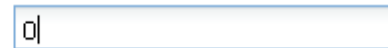
SDP bandwidth

A dropdown menu with a blue border. The top bar is blue and contains the text "enabled" in white. Below the bar, the menu is open, showing two options: "enabled" and "disabled", both in black text on a white background. A small blue arrow icon is visible on the right side of the top bar.

5. **SDP rate limit headroom (%)**—Enter the percentage of headroom to be added while using the AS bandwidth parameter while calculating the **Maximum bandwidth allowed per flow (bytes/sec)** (rate limit for the RTP flow). The default is 0. The valid range is:

- Minimum—0
- Maximum—100

SDP rate limit headroom (%)

A text input field with a blue border. It contains the number "0" in black text.

6. Click **Apply**.

About QoS Reporting

This section describes the Net-Net SBC QoS reporting. QoS reporting provides you with real-time evaluation of network and route performance. It lets you contrast internal domain and external domain performance and facilitates SLA verification and traffic engineering. Net-Net SBC QoS reporting is a measurement tool that collects statistics on Voice over IP (VoIP) call flows for SIP and H.323. To provide information, the Net-Net SBC writes additional parameters to the Remote Authentication Dial-in User Service (RADIUS) call record.

You can use QoS statistics for SLA customer reporting, fault isolation, SLA verification, and traffic analysis. The Net-Net SBC employs specialized hardware to inspect Real-Time Transport Protocol (RTP) and Real-Time Transport Control Protocol (RTCP) flows while maintaining wire-speed packet forwarding. QoS metrics are collected and reported on a per-session and per call-leg basis. These metrics are reported through real-time RADIUS records along with call accounting data.

Overview

When a conversation is established between two endpoints, two flows are present in each direction:

- RTP flow carries traffic between endpoints with a predictable packet arrival rate. The packets headers have sequence numbers that are used to determine whether packets are missing or lost.
- RTCP flow carries information about the RTP flow and keeps a different record. The RTCP packets contain timestamps based on Network Time Protocol (NTP).

QoS Statistics

Reported QoS data includes the following per-flow statistics:

- RTP and RTCP lost packets—Count of lost packets for both RTP and RTCP based on comparing the sequence numbers since the beginning of the call or the last context memory poll.
- RTP and RTCP average jitter—Incremental number of packets for both RTP and RTCP that have been used to generate the total and max jitter since the beginning of the call or the last context memory poll. The incremental accumulated jitter (in milliseconds) over all the packets received.
- RTP and RTCP maximum jitter—Maximum single jitter value (in milliseconds) for both RTP and RTCP from all the packets since the beginning of the call or the last context memory poll.
- RTCP average latency—Number of RTCP frames over which latency statistics have been accumulated and the incremental total of latency values reported since the beginning of the call or the last context memory poll.
- RTCP maximum latency—Highest latency value measured since the beginning of the call or the last context memory poll.
- RTP packet count
- RTP bytes sent and received
- RTCP lost packets—RTP lost packets reported in RTCP packets.
- ATP lost packets—Lost packets determined by monitoring RTP sequence numbers.

RADIUS Support

All the QoS statistics go into the RADIUS CDR. If a RADIUS client is configured on the Net-Net SBC, any time a call occurs a record is generated and sent. Only Stop RADIUS records contain the QoS statistic information.

For additional details about see the Net-Net system's support for RADIUS, including a dictionary of the standard RADIUS attributes supported by the Net-Net system and the Acme Packet accounting VSAs, see the *RADIUS Reference Guide*.

The following example shows a CDR that includes QoS data:

```

Wed Jan 26 11:23:41 2005
NAS-Identifier = "other"
Acct-Status-Type = Accounting-On
NAS-IP-Address = 172.30.29.10
NAS-Port = 0
Acct-Session-Id = "acmesystem#13261"
Client-IP-Address = 172.30.29.10
Acct-Unique-Session-Id = "63854a98e051defe"
Timestamp = 1106756621

Wed Jan 26 11:23:56 2005
NAS-Identifier = "other"
Acct-Status-Type = Start
NAS-IP-Address = 127.0.0.100
NAS-Port = 5060
Acme-Session-Ingress-CallId = "call-973579917-4@127.0.0.100"
Acme-Session-Egress-CallId = "SD2824301-19bb93c5454edfba33e7a9b6928234ec-v3000i1"
Acme-Session-Protocol-Type = "SIP"
Calling-Station-Id = "<sip:5678@127.0.0.11>;tag=SD2824301-1c10712"
Called-Station-Id = "sip:1234@112.22.33.444"
h323-setup-time = "12:44:19.115 UTC JAN 26 2005"
h323-connect-time = "12:44:22.809 UTC JAN 26 2005"
Acme-Session-Egress-Realm = "core_1"
Acme-Session-Ingress-Realm = "peer_1"
Acme-FlowID = "localhost:65536"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "peer_1"
Acme-Flow-In-Src-Addr = 0.0.0.0
Acme-Flow-In-Src-Port = 0
Acme-Flow-In-Dst-Addr = 192.168.0.11
Acme-Flow-In-Dst-Port = 21000
Acme-Flow-Out-Realm = "core_1"
Acme-Flow-Out-Src-Addr = 172.16.0.11
Acme-Flow-Out-Src-Port = 31000
Acme-Flow-Out-Dst-Addr = 0.0.0.0
Acme-Flow-Out-Dst-Port = 0
Acme-FlowID = "localhost:65537"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "core_1"
Acme-Flow-In-Src-Addr = 0.0.0.0
Acme-Flow-In-Src-Port = 0
Acme-Flow-In-Dst-Addr = 172.16.0.11
Acme-Flow-In-Dst-Port = 31000
Acme-Flow-Out-Realm = "peer_1"
Acme-Flow-Out-Src-Addr = 192.168.0.11

```

```

Acme-Flow-Out-Src-Port = 21000
Acme-Flow-Out-Dst-Addr = 192.168.0.101
Acme-Flow-Out-Dst-Port = 8766
Client-IP-Address = 172.30.29.10
Acct-Unique-Session-Id = "9dd15fd6f2881477"
Timestamp = 1106756636

Wed Jan 26 11:24:13 2005
NAS-Identifier = "other"
Acct-Status-Type = Stop
NAS-IP-Address = 127.0.0.100
NAS-Port = 5060
Acme-Session-Ingress-CallId = "call-973579917-4@127.0.0.11"
Acme-Session-Egress-CallId = "SD2824301-19bb93c5454edfba33e7a9b6928234ec-v3000i1"
Acme-Session-Protocol-Type = "SIP"
Calling-Station-Id = "<sip:5678@127.0.0.11>;tag=SD2824301-1c10712"
Called-Station-Id = "sip:1234@127.0.0.17"
Acct-Terminate-Cause = User-Request
Acct-Session-Time = 18
h323-setup-time = "12:44:19.115 UTC JAN 26 2005"
h323-connect-time = "12:44:22.809 UTC JAN 26 2005"
h323-disconnect-time = "12:44:40.707 UTC JAN 26 2005"
h323-disconnect-cause = "1"
Acme-Session-Egress-Realm = "core_1"
Acme-Session-Ingress-Realm = "peer_1"
Acme-FlowID = "localhost:65536"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "peer_1"
Acme-Flow-In-Src-Addr = 192.168.0.101
Acme-Flow-In-Src-Port = 8766
Acme-Flow-In-Dst-Addr = 192.168.0.11
Acme-Flow-In-Dst-Port = 21000
Acme-Flow-Out-Realm = "core_1"
Acme-Flow-Out-Src-Addr = 172.16.0.11
Acme-Flow-Out-Src-Port = 31000
Acme-Flow-Out-Dst-Addr = 172.16.0.50
Acme-Flow-Out-Dst-Port = 8766
Acme-Flow-RTCP-Packets-Lost = 0
Acme-Flow-RTCP-Jitter = 15
Acme-Flow-RTCP-Latency = 0
Acme-Flow-RTCP-MaxJitter = 15
Acme-Flow-RTCP-MaxLatency = 0
Acme-Flow-RTP-Packets-Lost = 0
Acme-Flow-RTP-Jitter = 0
Acme-Flow-RTP-MaxJitter = 44
Acme-Flow-Octets = 233152
Acme-Flow-Packets = 850
Acme-FlowID = "localhost:65537"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "core_1"
Acme-Flow-In-Src-Addr = 172.16.0.50
Acme-Flow-In-Src-Port = 8766
Acme-Flow-In-Dst-Addr = 172.16.0.11

```

```

Acme-Flow-In-Dst-Port = 31000
Acme-Flow-Out-Realm = "peer_1"
Acme-Flow-Out-Src-Addr = 192.168.0.11
Acme-Flow-Out-Src-Port = 21000
Acme-Flow-Out-Dst-Addr = 192.168.0.101
Acme-Flow-Out-Dst-Port = 8766
Acme-Flow-RTCP-Packets-Lost = 0
Acme-Flow-RTCP-Jitter = 15
Acme-Flow-RTCP-Latency = 0
Acme-Flow-RTCP-MaxJitter = 15
Acme-Flow-RTCP-MaxLatency = 0
Acme-Flow-RTP-Packets-Lost = 0
Acme-Flow-RTP-Jitter = 0
Acme-Flow-RTP-MaxJitter = 6
Acme-Flow-Octets = 223020
Acme-Flow-Packets = 874
Client-IP-Address = 172.30.29.10
Acct-Unique-Session-Id = "9dd15fd6f2881477"
Timestamp = 1106756653

Wed Jan 26 11:24:29 2005
NAS-Identifier = "other"
Acct-Status-Type = Start
NAS-IP-Address = 127.0.0.100
NAS-Port = 5060
Acme-Session-Ingress-CallId = "call-973580641-3@127.0.0.17"
Acme-Session-Egress-CallId = "SD4gub301-
057e502db85f49a108fb70752e403cd4-v3000i1"
Acme-Session-Protocol-Type = "SIP"
Calling-Station-Id = "<sip:1234@127.0.0.17>;tag=SD4gub301-2c8158"
Called-Station-Id = "sip:5678@127.0.0.11"
h323-setup-time = "12:44:51.190 UTC JAN 26 2005"
h323-connect-time = "12:44:55.555 UTC JAN 26 2005"
Acme-Session-Egress-Realm = "peer_1"
Acme-Session-Ingress-Realm = "core_1"
Acme-FlowID = "localhost:65538"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "core_1"
Acme-Flow-In-Src-Addr = 0.0.0.0
Acme-Flow-In-Src-Port = 0
Acme-Flow-In-Dst-Addr = 172.16.0.11
Acme-Flow-In-Dst-Port = 31002
Acme-Flow-Out-Realm = "peer_1"
Acme-Flow-Out-Src-Addr = 192.168.0.11
Acme-Flow-Out-Src-Port = 21002
Acme-Flow-Out-Dst-Addr = 0.0.0.0
Acme-Flow-Out-Dst-Port = 0
Acme-FlowID = "localhost:65539"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "peer_1"
Acme-Flow-In-Src-Addr = 0.0.0.0
Acme-Flow-In-Src-Port = 0
Acme-Flow-In-Dst-Addr = 192.168.0.11
Acme-Flow-In-Dst-Port = 21002

```

```

Acme-Flow-Out-Realm = "core_1"
Acme-Flow-Out-Src-Addr = 172.16.0.11
Acme-Flow-Out-Src-Port = 31002
Acme-Flow-Out-Dst-Addr = 172.16.0.50
Acme-Flow-Out-Dst-Port = 8768
Client-IP-Address = 172.30.29.10
Acct-Unique-Session-Id = "9dd15fd6f2881477"
Timestamp = 1106756669

Wed Jan 26 11:24:37 2005
NAS-Identifier = "other"
Acct-Status-Type = Stop
NAS-IP-Address = 127.0.0.100
NAS-Port = 5060
Acme-Session-Ingress-CallId = "call-973580641-3@127.0.0.17"
Acme-Session-Egress-CallId = "SD4gub301-057e502db85f49a108fb70752e403cd4-v3000i1"
Acme-Session-Protocol-Type = "SIP"
Calling-Station-Id = "<sip:1234@127.0.0.17>;tag=SD4gub301-2c8158"
Called-Station-Id = "sip:5678@127.0.0.11"
Acct-Terminate-Cause = User-Request
Acct-Session-Time = 9
h323-setup-time = "12:44:51.190 UTC JAN 26 2005"
h323-connect-time = "12:44:55.555 UTC JAN 26 2005"
h323-disconnect-time = "12:45:04.217 UTC JAN 26 2005"
h323-disconnect-cause = "1"
Acme-Session-Egress-Realm = "peer_1"
Acme-Session-Ingress-Realm = "core_1"
Acme-FlowID = "localhost:65538"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "core_1"
Acme-Flow-In-Src-Addr = 172.16.0.50
Acme-Flow-In-Src-Port = 8768
Acme-Flow-In-Dst-Addr = 172.16.0.11
Acme-Flow-In-Dst-Port = 31002
Acme-Flow-Out-Realm = "peer_1"
Acme-Flow-Out-Src-Addr = 192.168.0.11
Acme-Flow-Out-Src-Port = 21002
Acme-Flow-Out-Dst-Addr = 192.168.0.101
Acme-Flow-Out-Dst-Port = 8768
Acme-Flow-RTCP-Packets-Lost = 0
Acme-Flow-RTCP-Jitter = 15
Acme-Flow-RTCP-Latency = 0
Acme-Flow-RTCP-MaxJitter = 15
Acme-Flow-RTCP-MaxLatency = 0
Acme-Flow-RTP-Packets-Lost = 0
Acme-Flow-RTP-Jitter = 0
Acme-Flow-RTP-MaxJitter = 38
Acme-Flow-Octets = 92688
Acme-Flow-Packets = 364
Acme-FlowID = "localhost:65539"
Acme-FlowType = "PCMU"
Acme-Flow-In-Realm = "peer_1"
Acme-Flow-In-Src-Addr = 192.168.0.101

```

```
Acme-Flow-In-Src-Port = 8768
Acme-Flow-In-Dst-Addr = 192.168.0.11
Acme-Flow-In-Dst-Port = 21002
Acme-Flow-Out-Realm = "core_1"
Acme-Flow-Out-Src-Addr = 172.16.0.11
Acme-Flow-Out-Src-Port = 31002
Acme-Flow-Out-Dst-Addr = 172.16.0.50
Acme-Flow-Out-Dst-Port = 8768
Acme-Flow-RTCP-Packets-Lost = 0
Acme-Flow-RTCP-Jitter = 15
Acme-Flow-RTCP-Latency = 0
Acme-Flow-RTCP-MaxJitter = 15
Acme-Flow-RTCP-MaxLatency = 0
Acme-Flow-RTP-Packets-Lost = 0
Acme-Flow-RTP-Jitter = 0
Acme-Flow-RTP-MaxJitter = 6
Acme-Flow-Octets = 113100
Acme-Flow-Packets = 428
Client-IP-Address = 172.30.29.10
Acct-Unique-Session-Id = "9dd15fd6f2881477"
Timestamp = 1106756677
```

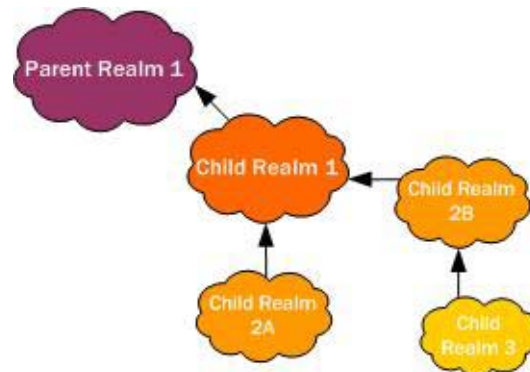

Aggregate Session Constraints: Nested Realms

In addition to setting session constraints per realm for SIP and H.323 sessions, you can also enable the Net-Net SBC to apply session constraints across nested realms. When you set up session constraints for a realm, those constraints apply only to the realm for which they are configured without consideration for its relationship, either as parent or child, to any other realms.

You can also, however, enable the Net-Net SBC to take nested realms into consideration when applying constraints. For example, if a call enters on a realm that has no constraints but its parent does, then the constraints for the parent are applied. This parameter is global and so applies to all realms on the system. For the specific realm the call uses and for all of its parents, the Net-Net SBC increments the counters upon successful completion of an inbound or outbound call.

In the following example, you can see one parent realm and its multiple nested, child realms. Consider applying the following realm constraints:

- Parent Realm 1—55 active sessions
- Child Realm 1—45 active sessions
- Child Realm 2A—30 active sessions
- Child Realm 2B—90 active sessions
- Child Realm 3—20 active sessions



Given the realm constraints outlined above, a call enters the Net-Net SBC on Child Realm 2B, which has an unmet 90-session constraint set. The Net-Net SBC allows the call based on Child Realm 2B. But the call also has to be within the constraints set for Child Realm 1 and Parent Realm 1. If the call fails to fall within the constraints for either of these two realms, then the Net-Net SBC rejects the call.

Impact to Other Session Constraints and Emergency Calls

You can set up session constraints in different places in your Net-Net SBC configuration. Since session agents and SIP interfaces also take session constraints, it is important to remember the order in which the Net-Net SBC applies them:

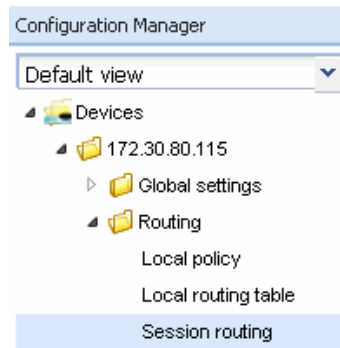
1. Session agent session constraints
2. Realm session constraints (including parent realms)
3. SIP interface session constraints

Emergency and priority calls for each of these is exempt from session constraints. That is, any call coming into the Net-Net SBC marked priority is processed.

Configuring Session Constraints for Nested Realms

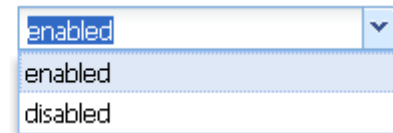
To configure session constraints for nested realms across the entire system:

1. Expand the Configuration Manager slider > Routing.



2. Click Session routing.
3. Click **Add** to add the session routing configuration in the content area.
4. **Apply constraints to all related realms**—Select enabled from the combination box to enable session constraints across all nested realms.

Apply constraints to all related realms



5. Click **Apply**.

Configuring QoS

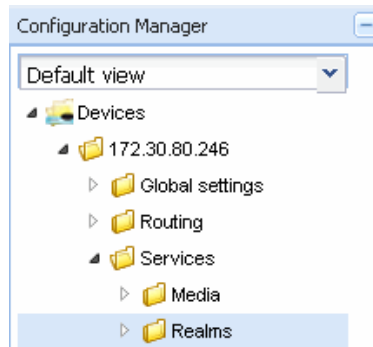
This section explains how to configure QoS. To generate QoS metrics, you need to enable QoS for the realm of the originating caller. The ingress realm determines whether QoS is turned on for a specific flow.

Note: If you run with QoS turned on one side only and disabled on the other you lose the ability to measure latency through the use of RTCP timestamps.

Configuring QoS

This section describes how to enable QoS on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure QoS for and click **Edit**. The Edit Realm window appears.
4. Scroll to QoS.
5. **QoS measurements**—Select enabled from the combination box to enable QoS.
6. **Profile name**—Select a preconfigured QoS Profile name from the combination box.

QoS

QoS measurements

enabled ▼

Profile name

▼
mp2

7. Click **Apply**.

Configuring Network Management Controls

The Net-Net SBC supports network management controls for multimedia traffic specifically for static call gapping and 911 exemption handling. These controls limit the volume or rate of traffic for a specific set of dialed numbers or dialed number prefixes (destination codes).

In TDM networks, automatic call/code gapping was developed as part of the advanced intelligent network (AIN) to enable network element load shedding based on destination number (DN) in case of overload. However, since there are as yet no standards for call/code gapping for next generation multimedia networks, the Net-Net SBC provides statically-provisioned network management controls.

How It Works

To enable network management controls on your Net-Net SBC, you configure the controls rules and then enable those rules on a per-realm basis. Each network management control rule has a unique name, in addition to information about the destination (IP address, FQDN, or destination number or prefix), how to perform network management (control type), whether to reject or divert the call, the next hop for routing, and information about status/cause codes. Details about the content of control rules as how to set them appear in the instructions and examples section.

When a SIP INVITE or an H.323 Setup for a call arrives at the Net-Net SBC on an ingress realm where network management controls have been enabled, the Net-Net SBC takes the following steps:

- It searches the network management rules you have created (and which are stored in tables on the Net-Net SBC) for a control rule that best matches the newly-received call.
- If it does not find a matching control rule, the Net-Net SBC allows the call to proceed normally.
- If it finds a matching control rule, then the Net-Net SBC treats the call according to the specifics of the rule and the treatment method that it designates.

Matching a Call to a Control Rule

The Net-Net SBC uses the call classification key to match calls so that it can apply control rules. The call classification key specifies information about the destination, which can be an IP address, an FQDN, a destination (called) number, or destination prefix. You configure the classification key as part of the control rule.

Matching is performed from left to right, starting at the left-most character. A wildcard matches any digit.

The Net-Net SBC compares the following information from the SIP INVITE or H.323 Setup for matching:

- SIP INVITE—User part of the Request URI, or the host part of the Request URI
- H.323 Setup—Q.931 Called Party Number IE

If multiple rules match an incoming call, then the Net-Net SBC gives first priority to destination number or the destination prefix. Next, it tries to match based on the IP address, and finally it looks to the domain (lowest priority).

How It Works: IWF Calls

For calls that require interworking between SIP and H.323, the Net-Net SBC performs call management control on the ingress leg of the call. If the call matches a control rule, the Net-Net SBC applies the treatment rule at the time it processes the ingress call. In addition:

- When the treatment method is rejection, the Net-Net SBC immediately rejects the call on the ingress leg.
- When the treatment method is call diversion, the Net-Net SBC sends the call to its SIP or H.323 task for completion, and this task does not repeat network management control rule application.

Before sending on the call to its SIP or H.323 task, the Net-Net SBC adds the `acme_iwf_nmc=<nmc_name>` Request URI parameter to the Request URI (where `nmc_name` is the name of a control rule). When the INVITE is received and that parameter is found, the Net-Net SBC applies the specified control rule to the session.

- When the control rule type is priority, the Net-Net SBC's SIP or H.323 process adds a request URI parameter that flags the call as priority before it sends an INVITE to either the SIP or H.323 process. When the INVITE is received and the priority request URI is found, the Net-Net SBC makes the call and bypasses any local network management.

Call Handling Determination

There are three types of control rules from which you can choose; each is a different way for the Net-Net SBC to handle calls matching the classification key:

- Call gap rate—Controls the maximum sustained rate of calls that match the classification key.

Using this type, the Net-Net SBC calculates the time since the last matching call. If that time is equal to or greater than the minimum time gap set in the control rule (i.e., it does not exceed the rate), then the call proceeds normally. If the call is less than the minimum time gap (i.e., it causes the call rate to be exceeded), then the Net-Net either rejects or diverts the call.

To keep the call rate below the control value, the Net-Net SBC ensures a minimum call gap time between the matching calls. For example, if the control value is 10 calls per second, the minimum call gap time would be 0.1 second. And if a matching call were to arrive within a tenth of a second since the last matching call, then the Net-Net SBC applies the treatment method.

- Call gap percentage—Controls the percentage of calls matching the classification key you set for the control rule.

When using this control rule type, the Net-Net SBC applies the treatment method to the percentage of matching calls (that you set in the value parameter) out the total number of matching calls it receives. For example, if you set the value parameter for the control rule to 50 and use this control type, the Net-Net SBC applies the treatment method to every other call it receives (or 50% of the calls it receives) that matches the classification key.

Note that the Net-Net SBC cannot maintain exact percentages for the control value at all times, especially at system start-up when the number of incoming calls is small.

- Priority—Exempts calls to a destination (like 911) from local network management controls such as:
 - Session agent constraints
 - Bandwidth constraints (such as per-realm bandwidth)
 - External policy servers (requests are made to the policy server; calls are admitted and processed regardless of the decision or reachability of the policy server)
 - Per-user call admission control
 - CPU constraints

The Net-Net SBC will not bypass licensing constraints, however.

Treatment Methods

You can choose from two different treatment methods:

- Call rejection—The Net-Net SBC rejects the call.
 - For SIP, the Net-Net SBC sends a response messages with the status code of the control rule. This response message also includes a Reason header with the Q.850 cause code that you configure as part of the control rule; it contains only the Q.850 cause code, and there is no reason-text included. For example:
 Reason: Q.850; cause=63
 - For H.323, the Net-Net SBC sends a releaseComplete message with the Q.850 cause code (that you configure as part of the control rule) of the control rule as the Q.931 Cause IE.

- Call diversion—The Net-Net SBC routes the call to the location you specify in the control rule's next hop parameter.

Except for this routing, the call proceeds as normal. Local treatments such as number translation apply to the call, as do local controls such as licensing. Note the following:

- If the next hop is an FQDN, the Net-Net SBC performs DNS queries to resolve the next hop to an IP address so that it can route the call properly.
- DNS queries only apply to pure SIP or IWF calls that originate in H.323 and are interworked to SIP.
- If the next hop is a session agent group, the Net-Net SBC selects a session agent from the group according to the selection strategy you set for the group. Then the Net-Net SBC uses the IP address of the selected session agent.

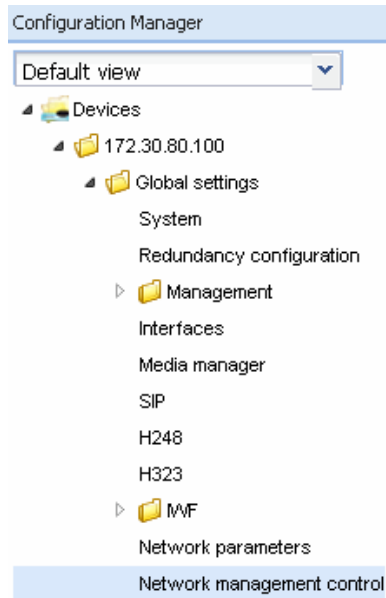
Configuring Network Management Controls

Creating an NM Control Rule

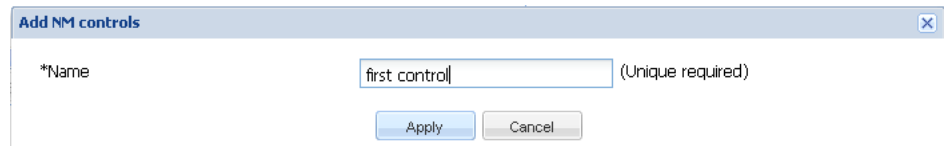
In order use the network management controls feature, you need to set control rules and then enable their application on a per-realm basis. This section shows you how to set up those configurations.

To create an NM control rule:

1. Expand the Configuration Manager slider > Global settings.



2. Expand the Global settings folder and click Network management control.
3. Click **Add**. The Add NM controls dialog box appears.
4. **Name**—Enter a name of this network management control rule; this value uniquely identifies the control rule. Click **Apply**.



5. Select the control rule from the table and click **Edit**.
6. **State**—Select enabled from the combination box.
7. **Type**—Select a control type from the combination box or leave this parameter blank.

The following table lists the control types available:

Control Type	Description
gap-rate	Controls the maximum sustained rate of calls that match the classification key.

Control Type	Description
gap-percent	Controls the percentage of calls matching the classification key you set for the control rule.
priority	Exempts calls to a destination (like 911) from local network management controls.

For more information about control types, refer to the Call Handling Determination section above.

8. **Value**—If you set the control type to either gap-rate or gap-percent, you need to enter the maximum sustained rate of calls or the percentage of calls to which you want the control rule applied.

The default is zero, and:

- For the **gap-rate**, the maximum is 2147483647 (which you can set by entering -1 as the value, an option provided for ease of use)
 - Using the minimum value (0) means that the Net-Net SBC treats all calls
 - Using the maximum value means that the Net-Net SBC treats no calls
- For the **gap-percent**, the maximum is 100
 - Using the minimum value (0) means that the Net-Net SBC treats no calls
 - Using the maximum value (100%) means that the Net-Net SBC treats all calls

9. **Treatment**—Select a method from the combination box or leave this parameter blank.

The following table lists the treatment methods.:

Treatment	Description
apply-local-policy	
reject	The Net-Net SBC rejects the call.
divert	The Net-Net SBC routes the call to the location you specify in the control rule's next hop parameter.

For more information about treatment methods, refer to the Treatment Methods section above.

10. **Next hop**—Type the next hop for the Net-Net SBC to use when the treatment method is **divert**.

This value should contain one of the following:

- hostname(:port)
- IP address(:port)
- Name of a valid, configured session agent
- Name of a valid, configured session agent group—When you set this parameter to a session agent group, you must specify that it is a session agent group by prepending the name of the group with either **SAG:** or **sag:**. For example, the entry for a session agent group with **Group2** as its name would be **SAG:Group2** or **sag:Group2**.

11. **Realm next-hop**—Select the realm of the next hop from the combination box when the treatment type is **divert**.
12. **Protocol next-hop**—Select the signaling protocol for the next hop from the combination box when the treatment type is **divert**.
13. **Status code**—Enter the SIP response code that you want the Net-Net SBC to use when the treatment method is **reject**, or retain the default value **503** (Service Unavailable). The value you enter must be numeric, but it cannot be a 0.
14. **Cause code**—Enter the Q.850 cause code that you want the Net-Net SBC to use when the treatment method is **reject**. You can leave this parameter set to its default, 63 (Service or option not available) or set it to any numeric value except 0.
 - For a SIP call, the Net-Net SBC replaces the cause code in the Reason header of the SIP response.
 - For a H.323 call, the Net-Net SBC converts the cause code to a Q.931 cause code in the Q.931 Cause IE in the releaseComplete message.

State	enabled
Type	gap-percent
Value	1000
Treatment	divert
Next hop	172.30.10.100
Realm next-hop	realm1
Protocol next-hop	SIP
Status code	503
Cause code	63

Configuring the Destination ID

1. In the NM controls table, select a control rule and click **Edit**.
2. For Destination ID, click **Add**. The Add Destination ID dialog box appears.
3. **Destination ID**—Enter the classification key.

Add Destination ID

*Destination ID

192.40.10.110

(Unique required)

Apply

Cancel

This parameter specifies information about the destination, which can be an IP address, an FQDN, a destination (called) number, or destination prefix. You can wildcard characters in the classification key using the carat symbol (^).

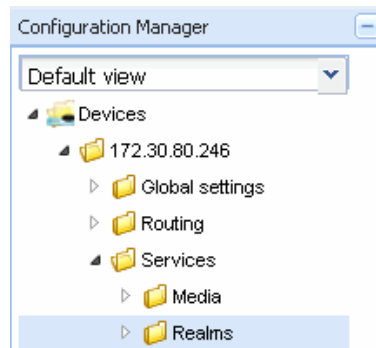
4. Click **Apply**. The Add Destinationid dialog box closes and the destination ID appears in the list on the Edit NM controls dialog box.

- Click **Apply**.

Applying NM Control Rule to a Realm

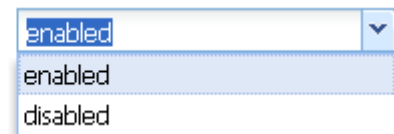
To apply an NM control rule to a realm:

- Expand the Configuration Manager slider > Services.



- Click **Realms**. The Realm table appears in the content area.
- Select the realm on which you want to apply NM control and click **Edit**.
- Apply network management (NM) controls—Select enabled from the combination box to enable it.

Apply network management (NM)
controls



- Click **Apply**.

Network Management Control Dialed Number Enhancement

Matching a Call to a Control Rule

The Net-Net SBC uses the call classification key, specified by the **destination-identifier**, to match calls so that it can apply control rules. The call classification key specifies information about the destination, which can be an IP address, an FQDN, a destination (called) number, or destination prefix. You configure the classification key as part of the control rule.

Matching is performed from left to right, starting at the left-most character. A wildcard matches any digit.

The Net-Net SBC compares the following information from the SIP INVITE or H.323 Setup for matching:

- SIP INVITE—User part of the Request URI, or the host part of the Request URI
- H.323 Setup—Q.931 Called Party Number IE

The Net-Net SBC normalizes the user-part of the Request-URI prior to performing any matching for NMC based on the dialed number. This normalization strips out any of the visual-separator characters. For example:

- +358-555-1234567 after normalization is: +3585551234567
- +(358)555.1234567 after normalization is: +3585551234567

Normalization occurs only for NMC look-up purposes, and it does not alter the actual Request-URI. If multiple rules match an incoming call, then the Net-Net SBC gives first priority to destination number or the destination prefix. Next, it tries to match based on the IP address, and finally it looks to the domain (lowest priority).

Specifically, the Net-Net SBC supports the following:

- The user-part can contain escaped sequences that the Net-Net SBC normalizes to their unescaped representation. For example:
 - %23(358)555.1234567 after normalization is: #3585551234567.
- The Net-Net SBC parses the user-part of the Request-URI up to the first semicolon (;). For example, the user-part in tel:+358-555-1234567;postd=pp22 will be +358-555-12134567.
- You can use the caret (^) to wildcard digits 0-9 in the destination-identifier you configure.

You can enter and match special characters such as the plus-sign (+), the asterisk (*), the pound sign (#), capital letter A (A), capital letter B (B), capital letter C (C), capital letter D (D), capital letter E (E), capital letter F (F), lowercase letter p (p), lowercase letter w (w).

Configuring the Account Configuration and Account Servers

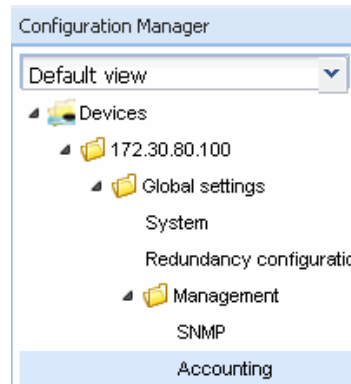
This section explains how to configure the account configuration and account servers so you can use the Net-Net system in conjunction with external RADIUS (accounting) servers to generate CDRs and provide billing services requires.

For more information about RADIUS, see the *Net-Net RADIUS Reference Guide*.

Configuring the Account Configuration and Account Servers

This section describes how to configure the account configuration and account servers on your Net-Net SBC.

1. Expand the Configuration Manager slider > Global settings > Management.



2. Click Accounting.
3. Press **Add** to add the accounting configuration in the content area.
4. **State**—Select the State of the account configuration from the combination box.
5. **Host name**—Enter the Host name of the account configuration.
6. **Port**—Enter the port number of the account configuration.
7. **Start**—Select the type of SIP event that triggers the Net-Net system to transmit a RADIUS Start message from the combination box.

State	enabled
Host name	LOCALHOST
Port	1813
Protocol	RADIUS

Messages

Start	OK

8. **Interim**—Click **Add** in the Interim table. Select the type of SIP event that triggers the Net-Net system to transmit a RADIUS Interim message in the Combination box.

- Click **Apply**.

- Transmission timing**—Select whether you want the Net-Net system to transmit accounting information only at the close of a session from the combination box.

Transmission timing

enabled

enabled

disabled

- Max time-message retransmit (secs)**—Enter the time, in seconds, the Net-Net system continues to send each accounting message.
- Max. message queue before failover (#)**—Enter the number of accounting messages to be held in the message waiting queue before a failover situation status is enacted.

Max time-message retransmit(sec) 60

Max. message queue before failover (#) 100

Account Server

This section describes how to configure accounting servers on your Net-Net system.

- Expand the Configuration Manager slider > Global settings > Management.

- Click Accounting.
- Press **Add** to add the accounting configuration in the content area.
- Click the Radius accounting tab at the top of the content area.
- Host name or IP address**—Enter the name or IP address of the host associated with the account server.
- IP port**—Retain the default **1813** or enter the number of the DUP port associated with the account server to which RADIUS messages are sent.
- Click **Apply**.
- Select the server from the table and click **Edit**.

9. **State**—Select the state of the account server on the Net-Net system from the combination box. The default is enabled.
10. **Min. round trip time threshold (ms)**—Enter the minimum round trip time of an accounting message. The default is 250 milliseconds. If the fastest RTT is the strategy for the account configuration the value you enter here can be used to determine an order of preference (if all the configured account servers are responding in less than their minimum RTT).
11. **VSA bundling**—Select **enabled** from the combination box if you want the account server to bundle the VSAs within RADIUS accounting messages. Select **disabled** if you do not. (Bundling means including multiple VSAs within the vendor value portion of the message.)
12. **Max. time ACK (sec)**—Enter the length of time in seconds that you want the Net-Net system with pending accounting messages to wait when it has not received a valid response from the target account server. The default is **60** seconds.
13. **Min. time-reinstate delay (sec)**—Enter the length of time in seconds that you want the Net-Net system to wait before resending messages to a disabled account server. The default is **30** seconds.

State	enabled
VSA bundling	enabled
*Host name or IP address	BEDFORD1
*IP port	1813
Min. round trip time threshold (ms)	250
Max.time ACK (sec)	60
Min.time-reinstate delay (sec)	30
Priority	0

14. **Shared secret**—Enter the secret password from the account server to the client. Transactions between the client and the the RADIUS server are authenticated by the shared secret.
15. **NAS ID**—Enter the NAS ID. The account server uses this value to identify the Net-Net system for the transmittal of accounting messages.

Shared secret	*****
NAS ID	4321

16. Click **Apply** to finish configuring the accounting server.

Aggregate Session Constraints Per Realm

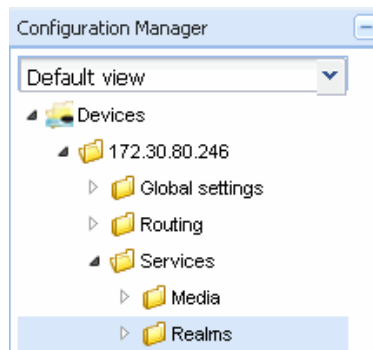
You can set session constraints for the Net-Net SBC's global SIP configuration, specified session agents, and specified SIP interfaces. Setting constraints forces users who have a large group of remote agents to create a large number of session agents and SIP interfaces.

You can also set session constraints for a specific realm. You can group remote agents into one or more realms upon which to apply session constraints. These constraints apply to all SIP and H.323 sessions when enabled.

Configuring Session Constraints per Realm

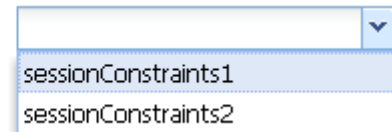
To enable sessions constraints on a per realm basis:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm from the table and click **Edit**.
4. **Session constraint name**—Select the name of the constraint you want to use from the combination box.

Session constraint name

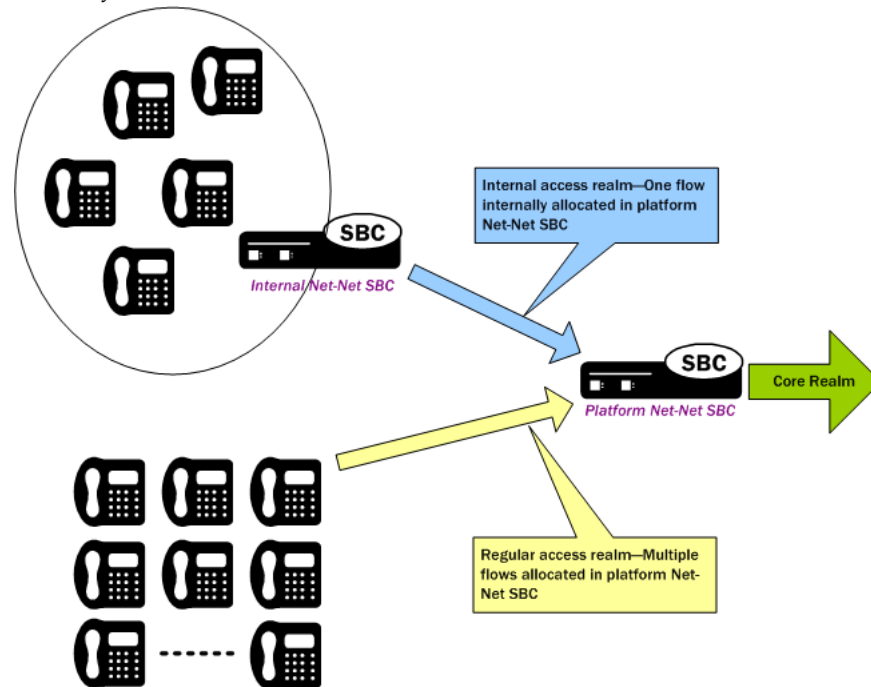


5. Click **Apply**.

Session Agent Minimum Reserved Bandwidth

You can assign session agents minimum bandwidth in access Net-Net SBC deployments. Assigning a session agent minimum bandwidth can prevent overloading other network devices—such as another Net-Net SBC configured as a session agent. Doing so ensures signaling bandwidth and availability to the endpoints behind this Net-Net SBC.

In the following diagram, the internal Net-Net SBC is configured as a session agent on the platform Net-Net SBC (which conveys traffic to the core realm). Setting up bandwidth reservation allows for the creation of only one allocated flow, and secures bandwidth for all the SIP clients behind the internal Net-Net SBC. Contrast this scenario with the one where the platform Net-Net SBC must allocate multiple flows for many SIP clients.



How It Works

When you configure minimum reserved bandwidth for session agent to a non-zero value, the Net-Net SBC allocates a separate pipe for per session agent. This is achieved by setting up an access control configuration in a specific way, instructing the Net-Net SBC to use a minimum number of transmission timeslots the individual pipe is guaranteed to receive.

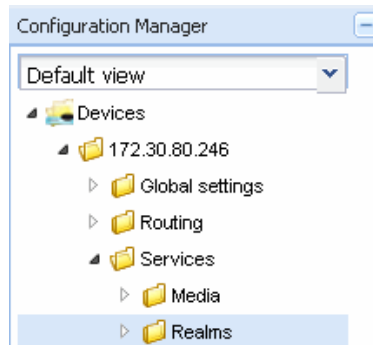
This feature works across all signaling services: SIP, H.323, and MGCP. No more than 4000 session pipes are supported.

Configuring Minimum Reserved Bandwidth For Session Agents

For the feature to work, you must set up an access control configuration with the settings required as shown in the instructions and examples below.

To configure minimum reserved bandwidth for session agents:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure in the realm table and click **Edit**.
4. Click the Access control tab at the top of the content area.
5. Click a realm in the Access control table and click **Edit**. The Edit Access control window appears.
6. **Minimum reserved bandwidth (pbs)**—Enter the minimum reserved bandwidth you want for the session agent, and that will trigger the creation of a separate pipe for it. Only a non-zero value will allow the feature to work properly, along with the other required values set out in these instructions. The default is 0, and the maximum is 0xffffffff (or 4294967295).

Minimum reserved bandwidth (pbs)

4294967295

7. Click **Apply**.

Aggregate Session Constraints for SIP

You can configure a full suite of session constraints and apply them to a SIP interface.

Session constraints configured natively on a SIP interface take precedence over session constraints applied in aggregate to a SIP interface.

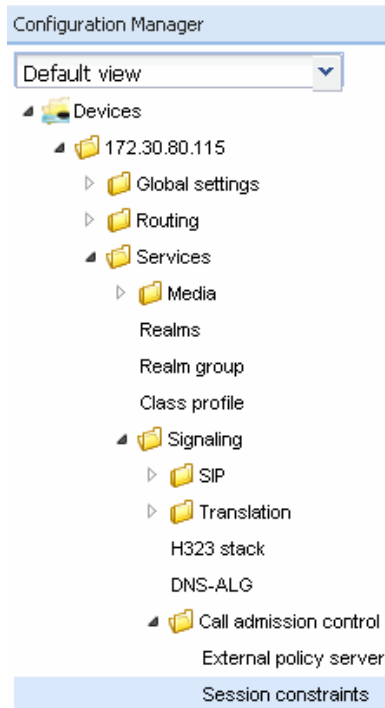
Using the constraints you have set up, the Net-Net SBC checks and limits traffic according to those settings for the SIP interface. If you do not set up the session constraints or you do not apply them in the SIP interface, then that SIP interface will be unconstrained.

SIP interfaces have two states: “In Service” and “Constraints Exceeded.” When any one of the constraints is exceeded, the status of the SIP interface changes to “Constrained”. The session constraint timers that apply to the SIP interface are: After threshold reached before resuming INVITES, Burst window (sec), and Sustained window.

Configuring Session Constraints

To configure the session constraints:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click Session constraints.
3. Click **Add**. The Add Session constraints window appears.
4. **Session constraints name**—Enter the name for this session constraints configuration; this is a unique identifier that you will use in the SIP interface when you want the session constraints applied there.
5. **State**—Select enabled from the combination box.
6. **Max. Number of sessions (#)**—Enter the maximum sessions allowed for this constraint. The minimum value is 0 (default), and the maximum is 999999999.
7. **Max. burst rate (# sessions per sec)**—Enter the maximum burst rate (invites per second) allowed for this constraint. This value should be the sum of the max. inbound burst rate and the max. outbound burst rate. The minimum value is 0 (default) and the maximum value is 999999999.
8. **Max. sustained rate (# sessions per sec)**—Enter the maximum rate of session invitations per second allowed within the current window for this constraint. The minimum value is 0 (default), and the maximum is 999999999.

For the sustained rate, the Net-Net SBC maintains a current and previous window size. The period of time over which the rate is calculated is always between one and two window sizes.
9. **Max. inbound sessions (#)**—Enter the maximum inbound sessions allowed for this constraint. The minimum value is 0 (default), and the maximum is 999999999.
10. **Max. inbound burst rate (# sessions per sec)**—Enter the maximum inbound burst rate (number of session invitations per second) for this constraint. The

minimum value for this parameter is 0 (default), and the maximum is 999999999.

11. **Max. inbound sustained rate (# sessions per sec)**—Enter the maximum inbound sustain rate (of session invitations allowed within the current window) for this constraint. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
For the sustained rate, the Net-Net SBC maintains a current and previous window size. The period of time over which the rate is calculated is always between one and two window sizes.
12. **Max. outbound sessions (#)**—Enter the maximum outbound sessions allowed for this constraint. The minimum value is 0 (default), and the maximum is 999999999.
13. **Max. outbound burst rate (# sessions per sec)**—Enter the maximum outbound burst rate (number of session invitations per second) for this constraint. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
14. **Max. outbound sustained rate (# sessions per sec)**—Enter the maximum outbound sustain rate (of session invitations allowed within the current window) for this constraint. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
15. **Burst window (sec)**—Enter the time in seconds that you want to use to measure the burst rate; the “window” is the time over which the burst rate is calculated, and is used for the over all burst rate as well as the inbound and outbound burst rates. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
16. **Sustained window**—Enter the time in seconds used to measure the sustained rate; the “window” is the time over which the sustained rate is calculated, and is used for the over all sustained rate as well as the inbound and outbound sustained rates. The minimum value for this parameter is 0 (default), and the maximum is 999999999.

*Session constraints name	constraints1
State	enabled ▼
Max. Number of sessions (#)	0
Max. burst rate (# sessions per sec)	0
Max. sustained rate (# sessions per sec)	0
Max. inbound sessions (#)	0
Max. inbound burst rate (# sessions per sec)	0
Max. inbound sustained rate (# sessions per sec)	0
Max. outbound sessions (#)	0
Maximum outbound burst rate (# sessions per sec)	0
Maximum outbound sustained rate (# sessions per sec)	0
Burst window (sec)	0
Sustained window	0

17. Scroll to the Time-to-wait timers section.
18. **After threshold reached before resuming INVITES**—Enter the number of seconds that is used to place an element (like a session agent) in the standby state when it has been take out of service because of excessive transaction timeouts. The minimum value for this parameter is 0 (default), and the maximum is 999999999.
19. **After being unresponsive before resuming INVITES**—Enter the time delay in seconds the SIP proxy must wait from the time that it sends an invitation to a session-agent and gets no response before it tries again to elicit a response from a previously unresponsive session agent. The minimum value for this parameter is 0 (default) and the maximum is 999999.
20. **After communication established before being considered 'in service'**—Enter the time in seconds the session-agent must be operational (once communication is re-established) before the session agent is declared to

be in-service. This value gives the session agent adequate time to initialize. The minimum value for this parameter is 0 (default) and the maximum is 999999.

Time-to-wait timers (sec)

After threshold reached before resuming INVITES	0
After being unresponsive before resuming INVITES	0
After communication established before being considered 'in service'	0

21. Scroll to the Answer seizure ratio section.
22. **Minimum number of seizures**—Enter the minimum number of seizures for a no-answer scenario. The default is 5 and the valid range is 0-999999.
23. **Minimum answer to seizure ratio percentage**—Enter the minimum answer to seizure ratio in percentage format. The minimum value for this parameter is 0 (default), and the maximum is 999999.

Answer seizure ratio

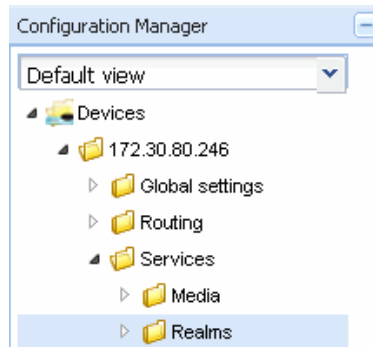
Minimum number of seizures	5
Minimum answer to seizure ratio percentage	0

24. Click **Apply**.

Applying Session Constraints to SIP Interfaces

To apply session constraints to a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to Session constraints.

7. **Session constraints**—Choose the name of the session constraints configuration that you want to apply to this SIP interface from the combination box.

Session constraint name

	▼
sessionConstraints1	
sessionConstraints2	

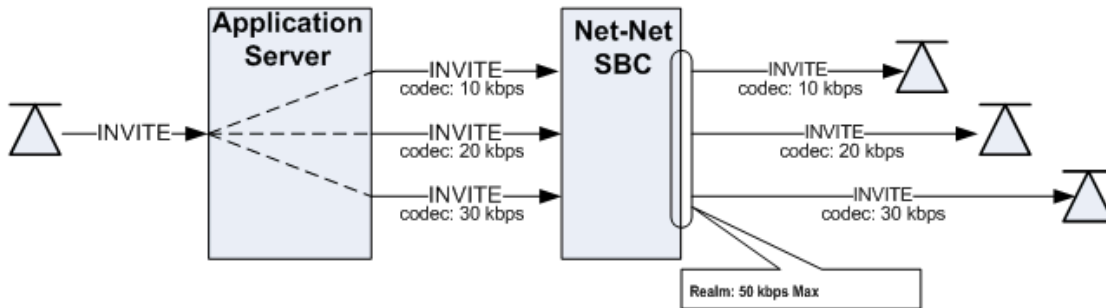
8. Click **Apply**.

Shared CAC for SIP Forked Calls

A forked call is one which has multiple INVITEs for the same call. For example, if an application server in the provider core network forks a call attempt, the application server sends several INVITEs for the same call toward the Net-Net SBC. Each INVITE is destined for a unique device that belongs to the same user. Ideally, that user will only answer one device. The Net-Net SBC treats each INVITE as a unique call request.

By default, each of the multiple INVITE forks are checked against CAC bandwidth limits, and thus they each consume bandwidth resources when they are received, even though only one of the forks will succeed in establishing a permanent session. Therefore, for many operators the CAC behavior of the Net-Net SBC is too restrictive and results in rejected call attempts which should have been allowed.

The following diagram shows a forked call scenario. The total bandwidth counted against the realm is 60 kbps. If the realm has a bandwidth ceiling of 50 kbps, one of the INVITEs will be rejected.



You can, however, enable the system to enforce CAC limits only once for SIP forked calls as long as the calls are identified as such, meaning that they will use the same bandwidth resources. The Net-Net SBC counts the forked call's most bandwidth-hungry codec at the time it arrives at the Net-Net SBC. In the above diagram, with shared bandwidth for forked calls enabled, the Net-Net SBC counts 30 kbps against the realm's total bandwidth after that INVITE arrives, even after the first two INVITES have passed into the final realm.

Bandwidth Sharing Scenarios

The following table summarizes how bandwidth would be shared given certain ingress and egress realms with this feature enabled. For the bandwidth to be shared,

Call A and Call B will have the same forked Call-ID in the P-Multiring-Correlator header.

CALL A					
CALL B		Ingress Realm A	Egress Realm B	Ingress Realm C	Egress Realm D
	Ingress Realm A	bandwidth shared	N/A	bandwidth not shared	N/A
	Egress Realm B	N/A	bandwidth shared	N/A	bandwidth not shared
	Ingress Realm C	bandwidth not shared	N/A	bandwidth shared	N/A
	Egress Realm D	N/A	bandwidth not shared	N/A	bandwidth shared

Configuring Bandwidth Sharing

To enable bandwidth sharing or forked calls, you set the **Bandwidth state for forked sessions** parameter in the SIP profile configuration to shared. Although there are other parameters available in the SIP profile configuration, you only have to set the name and the **Bandwidth state for forked sessions** values to use this feature.

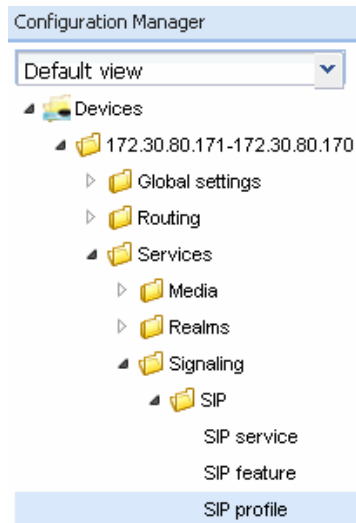
After you set up the SIP profile, you apply it to a realm, SIP interface, or session agent.

Configuring a SIP Profile

The SIP profile is an element in the Profiles configuration. You can configure multiple SIP profiles. Though this configuration contains additional parameters, you do not have to use them for the conditional bandwidth CAC for media release.

To create a SIP profile:

1. Expand the Configuration Manager slider > Services > Signaling > SIP.



2. Click SIP profile. The SIP profile table appears in the content area.
3. Click **Add**. The Add SIP profile dialog box appears.

4. **Name**—Enter the name of the SIP profile. You will use this name when you apply this profile to realms, session agents, and SIP interfaces. This parameter is required, and it has no default value.

The 'Add instance' dialog box has a title bar with a close button. Inside, the section is titled 'SIPPROFILE'. There is a label '*Name' followed by a text input field containing 'sipProfile1'. To the right of the input field is the text '(Unique required)'. At the bottom of the dialog are two buttons: 'Apply' and 'Cancel'.

5. Click **Apply**. The new SIP profile appears in the SIP profile table.
6. Select the SIP profile name and click **Edit**. The Edit SIP profile dialog box appears.
7. **Bandwidth state for forked sessions**—Click **shared** in the combination box if you want forked sessions to share bandwidth resources, or click **per-session** if you want bandwidth to be counted for each. Click **inherit** for an:
 - Ingress session agent without a SIP profile or with a SIP profile where the forked CAC mode is **inherit**, the Net-Net SBC will reference the associated realm.
 - Ingress realm without a SIP profile or with a SIP profile where the forked CAC mode is **inherit**, the Net-Net SBC will reference the associated SIP interface.
 - Ingress SIP interface without a SIP profile or with a SIP profile where the forked CAC mode is **inherit**, the Net-Net SBC will not perform bandwidth sharing for forked calls.

Bandwidth state for forked sessions

A dropdown menu showing four options: 'shared' (which is currently selected and highlighted), 'per-session', and 'inherit'. The menu is open, showing the list of options.

8. Click **Apply**.

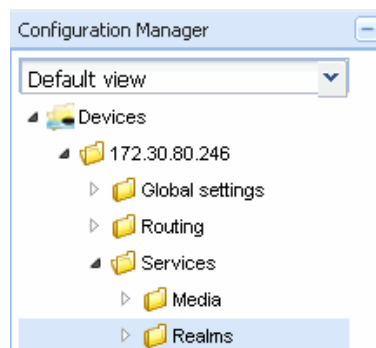
Applying a SIP Profile

Once you have configured one or more SIP profiles, you can apply them to realms, SIP interfaces, and SIP session agents.

Realms

To apply a SIP profile to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

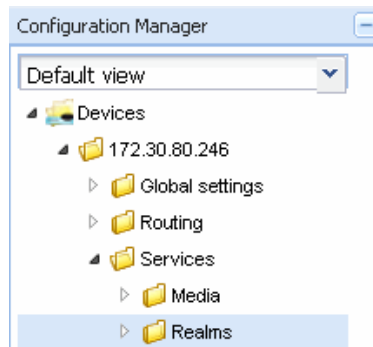
SIPprofile1

4. Click **Apply**.

SIP Interface

To apply a SIP profile to a SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**. The Edit Realm window appears.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**. The Edit SIP interface window appears.
6. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

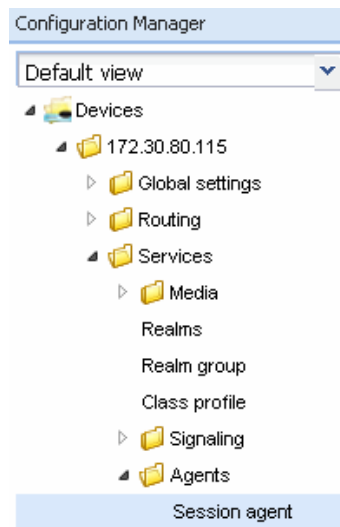
SIPprofile1

7. Click **Apply**.

Session Agent

To apply a SIP profile to a session agent:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. **SIP profile**—Click the name of the SIP profile that defines the redirection behavior in the combination box. This is the value you entered in the **name** parameter of the SIP profile configuration. This parameter has no default value.

SIP profile

SIPprofile1

4. Click Apply.

TLS Call Recording

2171

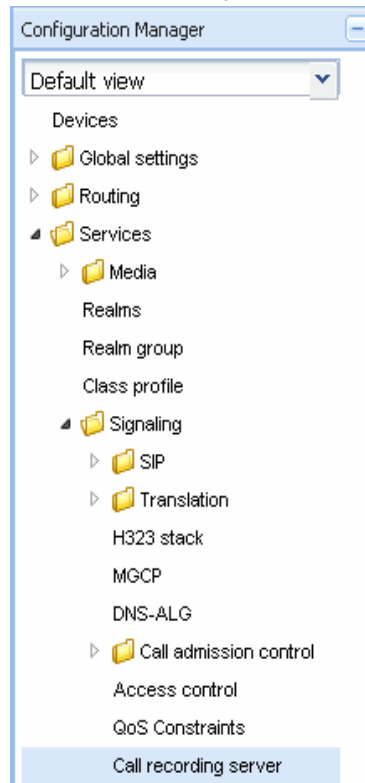
CONCEPTUAL

Configuring TLS Call Recording

To configure TLS call recording:

1. Expand the Configuration Manager slider -> Services -> Signaling -> Call admission control.

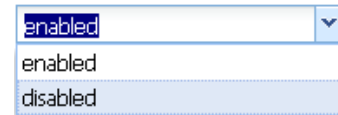
2. Click Call recording server.



3. Click the call recording server you want to configure in the table and click **Edit**.

4. **Decrypt TLS packets**—Click enabled in the drop-down list to enable

Decrypt TLS packets



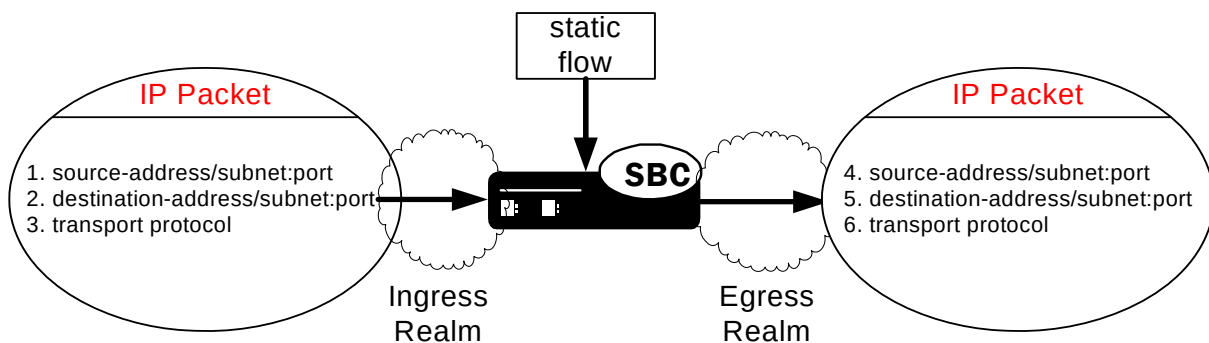
5. Click **Apply**.

Introduction

This chapter describes the Net-Net SBC's static flows feature. Static flows allow network traffic that matches specific criteria to pass through the Net-Net SBC unrestricted. This feature lets you steer traffic toward a particular destination based on its original characteristics. Static flows can range from being widely accessible to very restrictive, depending on the values you establish. Static flows are used for transporting a variety of signaling messages through the Net-Net SBC to achieve vendor interoperability.

About Static Flows

The static flow element explicitly writes entries into the IP routing table. These entries are persistent and are not deleted as calls are set up and broken down. Refer to the following diagram to understand how a static flow works.



A static flow entry watches for traffic with specific criteria on a specified ingress realm; that traffic consists of the following criteria:

1. The IPv4 packet enters the Net-Net SBC on the specified ingress realm.
2. The packet contains matching source address, subnet, and port criteria, field 1.
3. The packet contains matching destination address, subnet, and port criteria, field 2.
4. The packet contains a matching transport protocol, field 3.

If the above conditions are met, then the Net-Net SBC does the following:

1. The IPv4 traffic is forwarded out of the Net-Net SBC on the specified egress realm.
2. The configured source address, subnet, and port criteria are written to the exiting packet, field 4.
3. The configured destination address, subnet, and port criteria are written to the exiting packet, field 5.
4. The original transport protocol and its contents remain unchanged as the packet exits into the egress realm.

About Network Address Translation ALG

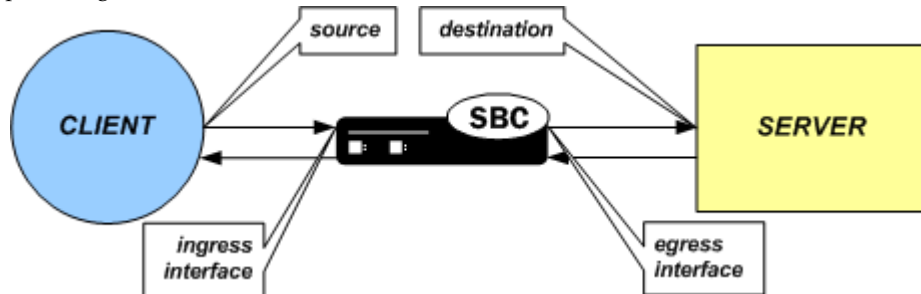
The Net-Net SBC supports Network Address and Port Translation (NAPT) and Trivial File Transfer Protocol (TFTP) functionality over media interfaces, collectively known as Network Address Translation (NAT) ALG. The NAT ALG feature is implemented as an extension of the static flow feature.

In some applications, the Net-Net SBC acts as an intermediary device, positioned between endpoints located in an access network and application servers located in a backbone network. The Net-Net SBC's NAT ALG feature enables these endpoints to use non-VoIP protocols, such as TFTP and HTTP, to access servers in a provider's backbone network to obtain configuration information.

NAT ALG parameters support RTC and can be dynamically reconfigured. The active NAT ALG configuration can be replicated on the standby SD in an HA configuration.

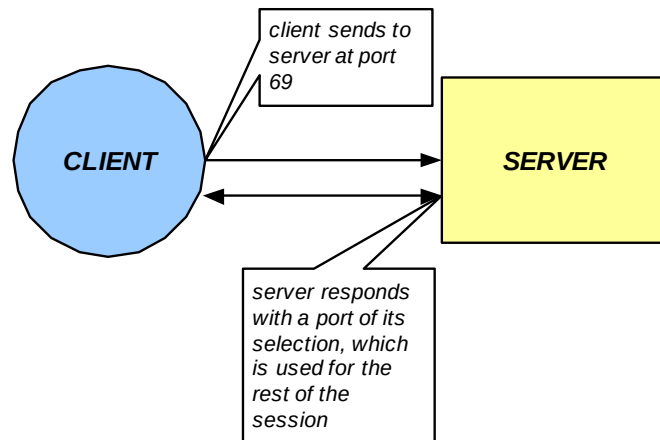
NAPT

The NAPT ALG functionality is the same as that found in commercially available enterprise and residential NAT devices. The Net-Net SBC watches for packets entering a media interface that match source and destination IP address criteria. Matching packets are then redirected out of the egress interface, through a specified port range, toward a destination address.



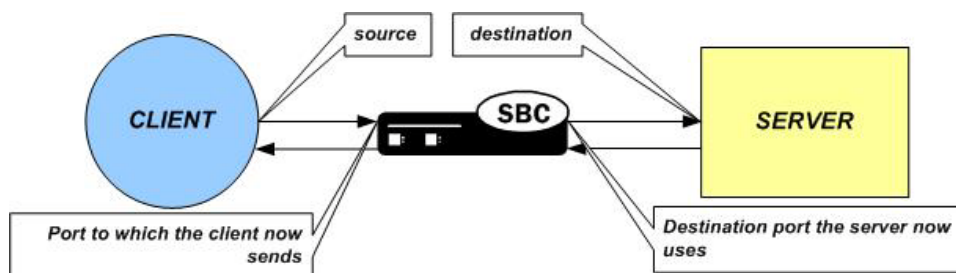
TFTP

The TFTP ALG is implemented as an extension of the NAT ALG. It works slightly differently than traditional NAPT. In a TFTP session, the first packet is sent from a source endpoint to port 69 on the TFTP server. The TFTP server responds from another port. This port, from which the TFTP response originates, is used for the remainder of the TFTP session.



To act as a TFTP ALG, the Net-Net SBC will latch on the first return packet from the server to learn the server's port. The ingress-side destination port of the Net-Net

SBC is changed to reflect the new communications port for the TFTP session. This process takes place without any user intervention.



Configuring Static Flows

This section explains how to configure static flows. It also provides sample configurations for your reference. You can configure static flows with or without NAT ALG. If you configure static flows with NAT ALG, you can choose NAPT or TFTP as the ALG type.

Basic Static Flow Configuration Overview

This section outlines the basic static flow configuration, without NAT ALG. You configure static flows by specifying ingress traffic criteria followed by egress re-sourcing criteria.

When configuring static flows, the following conventions are used:

- An IPv4 address of 0.0.0.0 matches all IPv4 addresses.
 - Not specifying a port implies all ports.
 - Not specifying a subnet mask implies a /32, matching for all 32 bits of the IPv4 address.
1. Set the static flows' incoming traffic-matching criteria. First set the ingress realm where you expect to receive traffic that will be routed via a static flow. Second, set the traffic's source IPv4 address, source subnet, and source port or port range criteria. Third, set the traffic's destination IPv4 address, destination subnet, and destination port criteria. This is usually an external address on the Net-Net SBC.
 2. Set the criteria that describes how traffic should be translated on the egress side of the Net-Net SBC. First set the egress realm where you want to send the traffic to be routed by this static flow. Second, set the traffic's source IPv4 address, source subnet, and source port or port range criteria. This is usually an external address on the Net-Net SBC. Third, set the traffic's destination IPv4 address, destination subnet, and destination port criteria.
 3. Set the protocol this static flow entry acts upon. This type of packet, as the payload of the IPv4 packet, remains untouched as traffic leaves the Net-Net SBC. Specifying a layer 4 protocol here acts as another criteria to filter against for this static flow.

The combination of entries in the ingress realm, ingress source address, ingress destination address, and protocol fields must be unique. For bidirectional traffic, you need to define a separate static flow in the opposite direction.

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by	admin@console
Last modified date	2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2_UTC

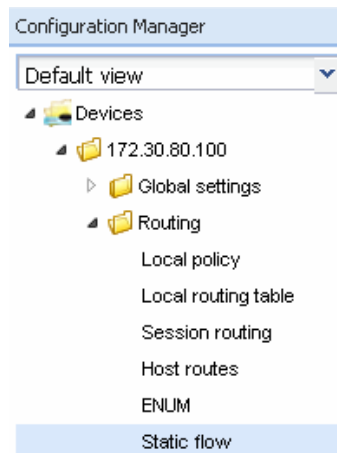
The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Configuring Static Flows

This section describes how to configure static flows and NAPT/TFTP ALG on your Net-Net SBC.

1. Expand the Configuration Manager slider > Routing.



2. Click Static flow. The Static flows table appears in the content area.
3. Click **Add**. The Add static flow dialog box appears.
4. Scroll to In destination.
5. **Port**—Enter the incoming destination IP address of packets to match for static-flow translation.

6. **IP Address**—Enter the incoming destination IP address of packets to match for static-flow translation.
7. **Number of bits**—Enter the In destination IP address. Enter the ingress side's destination. This is the IPv4 address and port of the network interface in the ingress realm.
8. Scroll to In source.
9. **Port**—Enter the port of the network that defines where the incoming traffic originates.
10. **IP Address**—Enter the IP address of the network that defines where the incoming traffic originates.
11. **Number of bits**—
12. **Realm**—Enter the ingress realm or interface source of packets to match for static flow translation. This in-realm-id field value must correspond to a valid identifier field entry in a realm-config. This is a required field. Entries in this field must follow the Name Format.
13. **Protocol**—Select the protocol this static flow applies to from the combination box.
14. Click **Apply** to continue. A confirmation window appears. Click the **OK** button to continue.

Add Static flow

In destination

*Port: 0

*IP Address: 172.16.1.16

*Number of bits: 0

In source

*Port: 0

*IP Address: 172.16.0.0

*Number of bits: 0

*Realm: in_sf

*Protocol: TCP

Apply Cancel

The new static flow appears in the Static flow window

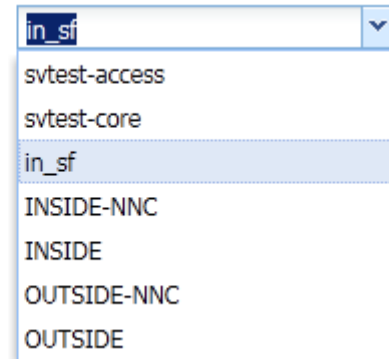
Editing Static Flows

1. Select the static flow you want to configure and click **Edit** in the Static flow window. The Edit static flow window appears.
2. **Description**—Enter an optional descriptive text string to identify the configuration and its function. Each description text box is blank by default.

From Realm

1. **Realm**—Click the access realm from the combination box where endpoints are located.

Realm



To Realm

2. **Realm**—Select the egress realm ID from the Realm field in the To Realm section of the screen.
3. Scroll to the Out source table and click Add.
4. **Port**—Enter the outgoing source IP address to translate to for static flow translation. Port 0 translates to packets sent on any port.
5. **IP Address**—Enter the outgoing source IP address of packets to translate to for static flow translation. IP address of 0.0.0.0 translates to any source address.
6. **Source (Address, Mask, Port)**—Enter the source network in the access realm where the endpoints exist. This parameter is entered as an IP address and netmask to indicate a range of possible IP addresses.
7. **Number of bits**—
8. Click **Apply** to continue.

Realms must be preconfigured in the realm object to appear in the combination boxes for this selection.

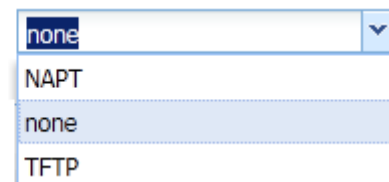
1. Scroll to the Out destination table and click Add.
2. **Port**—Enter the outgoing destination port of packets to translate to for static-flow translation. Port 0 translates to packets sent on any port.
3. **IP Address**—Enter the outgoing destination IP address of packets to translate to for static-flow translation. An IP address of 0.0.0.0 matches any source address.
4. **Number of bits**—
5. Click **Apply** to continue.

Realms must be preconfigured in the realm object to appear in the combination boxes for this selection.

Creating a NAPT or TFTP ALG Configuration

1. **ALG type**—Select either NAPT or TFTP from the combination box.

ALG type



2. **Maximum-bandwidth allowed per flow (bytes/sec)**—Enter the maximum bandwidth per flow for this ALG service.

Flow policing

Maximum bandwidth allowed per flow

3. **Start port**—Enter the starting port where the redirected flows are sourced from on the egress side.
4. **End port**—Enter the ending port where the redirected flows are sourced from on the egress side.
5. **Flow time limit**—Enter the total time in seconds for an ALG flow.
6. **Initial guard timer**—Enter the initial guard timer value in seconds for an ALG flow.
7. **Subsequent guard timer**—Enter the subsequent guard timer value in seconds for an ALG flow.

Application Layer Gateway

Start port

End port

Flow time limit

Initial guard timer

Subsequent guard timer

8. Click **Apply**.

Overview

High Availability (HA) enables continuous service by masking both planned and unplanned downtime and preventing single points of failure. Services remain up over 99% of the time in a given period. An HA cluster is a network of tightly linked servers that are also known as members or nodes. Fail-over clusters pooling together multiple members, each of which is a candidate server for your file systems, databases or applications. Each of these systems monitors the health of other systems in the cluster.

In the event of failure in one of the cluster members, the others take over the services of the failed node. When an interruption or failure occurs in a critical application, high availability clustering will combat this by switching the operations of this application to one of the other computers or nodes within that cluster.

Net-Net Central HA Cluster

Net-Net Central HA ensures that data and business processes are partitioned evenly across a cluster of NNC servers (members or nodes) without compromising availability. It ensures reliable access to NNC services at all times. Net-Net Central HA cluster of redundant nodes provides continuous service even if system components fail. In the cluster, the members work together to ensure that data remains continuously available. When a fault or potential interruption does occur, the HA cluster works to quickly and seamlessly prevent a complete system failure. Because there is not one single point of failure within the cluster, operations can continue without any noticeable interruptions to the user.

While Net-Net Central is based on a modular architecture that allows for customization, it is recommended that customers requiring a clustering solution run default configurations.

Terminology

Net-Net Central HA uses the following terminology:

- **HA:** Ability of a system to provide reliable access to services to its users. Highly available services remain up over 99% of the time in a given period.
- **Co-located HA:** System provides HA and high scalability using components that are geographically co-located.
- **Planned downtime:** Planned maintenance window that could be disruptive to system operation. Planned downtime events can include patches or re-configure components such as adding new members to a cluster or allowing members to rejoin a cluster.
- **Unplanned downtime:** Events that were not planned such as power outages, network, or hardware failures.
- **Election:** When the database master fails or becomes partitioned from the rest of the nodes in the cluster, an election is automatically held to elect a new database master. In a cluster composed of three or more nodes an election is won by the node with the simple majority of votes. Also, the node that wins an

election is always the one with the most recent log records. In the case of a tie, the node with the highest priority wins. In a two node cluster, an election can be won with a single vote. The replica can be elected master if the master fails.

HA Process

Net-Net Central (NNC) clustering offers reliable access to Net-Net Central services at all times. The cluster resilience in failure conditions provides the High Availability. The Net-Net Central cluster accomplishes this by categorizing data into different distribution mechanisms. The following summarizes these distribution mechanisms:

- **Device configuration:** The device is considered as the master database for configuration data. Any member of a cluster can go to the device and retrieve the most recent configuration. This process is called “on-demand” loading of configuration data. Not only is the configuration on each member synchronized to the same version, this has the advantage of removing the need to replicate large datasets between members.
- **Message driven data:** Some data sets that can be subject to network latency such as fault management events, polling statistics and audit trails are distributed via a Message-Oriented middleware (MOM). The MOM provides asynchronous processing that allows the NNC systems to scale both vertically and horizontally. The resilience of the MOM is maintained via guaranteed deliveries and durable subscribers. The data is generally stored on the local host machine in a dedicated local database.
- **Database replication:** Sensitive data sets such as LCV, user security, and device management are transactional and need to be available across the cluster are maintained by a database replication group. The database replication group maintains one master database in the cluster at all times while all other members are replicas. Retrieval of datasets is done on the local host machines. However, transactional modifications to the data are done on the master database, which then replicates the transaction to the replicas. Replication keeps the cluster database synchronized. If the master database fails, the replicas remaining elect a new master database.
- **Push-pull file transfer:** Large datasets such as route sets, that are maintained in the local database or file system are transferred around the cluster via push pull mechanisms. Host members use the MOM to publish events indicating that datasets are available. The other members use SFTP to pull or push the information from or to other members in the cluster.

The Net-Net Central HA process also guarantees execution of submitted tasks and ensures that data distribution is centralized. In order to ensure no single point of failure, a NNC server runs a load balancer, a front-end server, and a back-end server.

- **Load balancer:** Entrance point to Net-Net Central services. The load balancer provides SSL security (HTTPS), as well balancing loads among all front end servers contained in the cluster. All cluster members will run a load balancer to ensure that there is no single point of failure. Access to NNC services is not denied as long as one NNC server is running.
- **Front end server:** Responsible for maintaining the client interaction support. The front end server manages sessions and performs authentication and authorization. By default the front end server targets a local back end server.
- **Back end server:** Runs the services required to support any functionality provided by Net-Net Central. For example, the back end server can provide route management, fault management, or configuration management functionality.

The back end server also hosts the embedded database service and the message service. These services are responsible for maintaining the distributed data flow and provides for redundant failover capabilities.

- Embedded database service: Framework (Berkley DB XML) that provides the database XML support for the cluster. It is an in-memory database that supports replication and seamless negotiation of allocating a master database from among all the XML database members in the cluster. The database service (DBS) provides the database functionality in the back end server. Two database instances exist on any host, one is a local database XML and the other as part of a replication group.
- Message service: Message Oriented Middleware (MOM) used in the back end server to support distributed message queues and topics. MOM is supported by the distributed event and data service (DEDS) that allows components to publish and subscribe to message topics and queues.

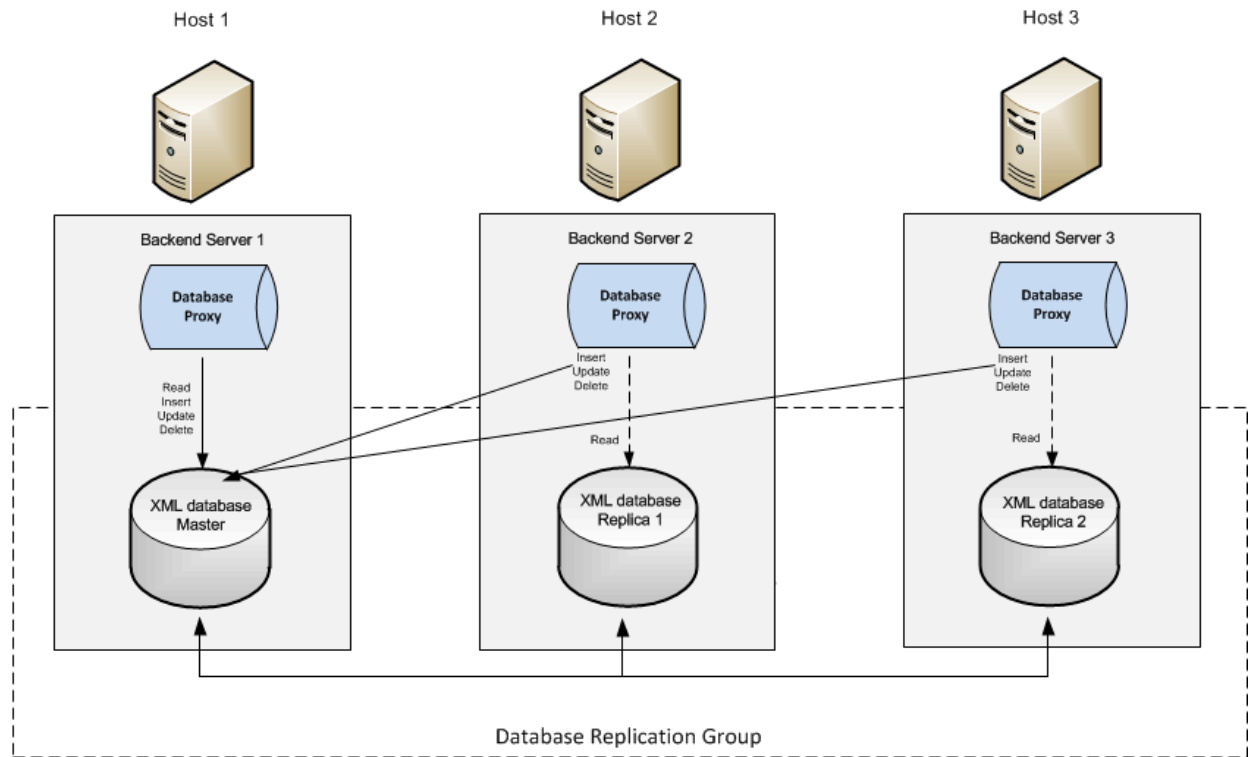
Single Master with Multiple Replica Strategy

NNC clustering offers two distinct types of clustering configurations these are:

- Multi-member cluster
- Two-member cluster

Multi-Member Clustering

In this strategy the NNC cluster contains three or more servers. Acme Packet recommends that for best HA performance and maintenance a three member cluster is configured as shown in the diagram below.



As illustrated in the diagram above, the cluster consists of three host systems. The database replication mechanism shown demonstrates the following points:

- Each host is running a backend server with an embedded-in-memory Berkeley XML database.
- In the replicated group, there is only one master database and multiple replicas.
- The master database is responsible for distribution of transactional modifications to the other replicas in the cluster.
- All back end server components interact with the database via a database proxy.
- The database proxy determines if the request for service is a transactional modification or a request for data retrieval. All data retrievals are done on the local database irrespective if it is a replica or master database. Requests for transactional modifications (inserts, updates or deletes) are forwarded from the database proxy to the master database in the cluster.

- The master database will guarantee the transactions on a quorum basis in a cluster. This means that in a cluster containing three or more members, the majority of the members need to reply that they have received the replicated datasets before the master returns success on the transaction. For example in a three member cluster the master needs to have a reply from at least one member before declaring success on the replication.
- User transactional latency is accounted for by detection of the late arrival of replicated data. Best effort replication is provided, which can mean the call might return before the dataset appears on the replicated databases. The database transactional layer offers additional support with latency in replicated data.

For example, a user on Host 3 starts a local transaction with the database proxy to insert content into the database. The database proxy in turn starts a transaction with the master database on Host 1. Each transaction that is started with the master database has a transactional id (txId) associated with it. The master database uses best effort in ensuring that the datasets are replicated to the other members of its replication group.

However, if the best effort takes longer than N seconds and the master database has received replies from quorum (the other replicas); the master database returns success. Returning success guarantees replication will occur at some point. The database Proxy on host 3 waits until the required txId appears in its local replicated database before returning success on the transaction to the user on host 3. This guarantees that the content inserted on the Master database has reached the replicated database. Users that initiate transactions are guaranteed to see the outcome of those transactions in their local database independent of which host the original transaction was initiated.

HA for Multi-Member Cluster

In a multi member cluster, high availability is provided for as follows:

- Database replication group:
 - Maintains one master in the entire cluster with multiple replicas.
 - On master database failure, re-election among the replicated database occurs and a new master database is elected.
 - For three or more member cluster, transactions are deemed successful only if quorum of replies from replicas is achieved. This guarantees that the data exists on more than the master database after the transaction completes. If quorum is not met the transaction will fail.
- Messaging event and data distribution:
 - The MOM distributes the messages in the cluster based on a store and forward process. The MOM guarantees message delivery by storing the message in a local database first before declaring that the message was properly processed.
 - There is no master in the MOM cluster, all MOM brokers participating in a cluster ensure that messages are synchronized in the cluster.
 - Durable subscribers ensure that even if a member leaves the cluster and reenters within a 24 hour period, missed messages are re-delivered.
 - Tasks entered in a queue are processed even if the host where the task was originally submitted goes down.
- On-demand device configuration loading:
 - The device is the master database for configuration data.

- No replication of large configurations is required.
- Failure of a member in the host does not effect on-demand retrieval of configuration from devices serviced by other members.

Database Group Replication Data Content

Data is not compromised and is guaranteed to be replicated throughout the cluster, which ensures that failover is seamless. The data includes:

- User management data, user credentials for authentication, and user access control lists (ACL) for authorization
- User configuration modifications made per device, known as the Local Configuration View (LCV). When users make modifications to the configuration on a targeted device, only the modification changes are maintained in the LCV. This smaller dataset is replicated, reducing the latency introduced when replicating larger datasets.
- Distributed locking data. This data indicates when an NNC lock has been placed on a targeted device for a specific operation. A distributed lock ensures that no two operations can occur concurrently on the same device.
- Net-Net SBC details created when a device is added to Net-Net Central for management. This dataset provides the information required for Net-Net Central to communicate with the targeted SBC as well as provide relevant hardware, firmware, configuration and reachability status.
- NNC configuration details. This dataset provides the user customized required configuration details that inform NNC services on how to function.
- Fault sequence number, which is a global unique ID required to give distributed items uniqueness.

Database Group Replication HA Behavior in Multi- Member Cluster

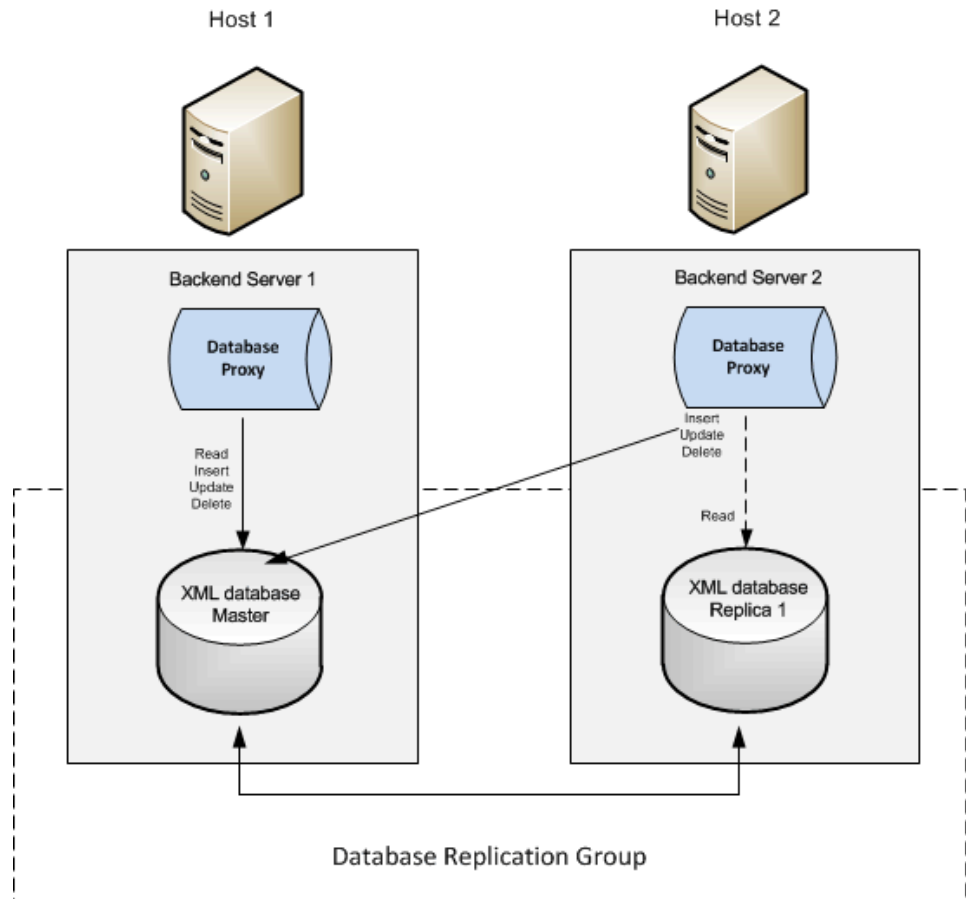
The following table lists the activities involved in database HA behavior in a multi-member cluster.

Activity	Description
Server startup	When an NNC server is started it joins the cluster as a replica and an election is held if there is currently no master. If the member ends up as a replica, then it is synchronized with the master during the initialization phase of the database service startup.
Master member failure	When the master fails or becomes partitioned from the rest of the members in the cluster, an election is automatically held by the remaining replicas to elect a new master.
Transactions (Quorum)	Transactions return successfully if a majority of the members in the cluster have replied that they received the replicated datasets. If quorum in replies from replicas is not achieved in a specific time period, the transaction fails.
Network partition	When a network partition exists between the members, only the members that can communicate with a majority of the members may elect a new master database. Members that can not communicate with a majority will enter READ-ONLY mode. Upon re-establishing network connectivity, re-elections take place and a master is elected while the other members revert to replicas. Nodes that were partitioned in READ-ONLY mode will automatically shut down to clean up stale database handles.

Activity	Description
Elections	In a cluster of three or more members, an election is won by the member with the simple majority of votes. The member that wins an election is always the one with the most recent log records. In the case of a tie, the member with the highest priority wins.
Rejoining a cluster after a graceful shutdown	When a server is restarted within several hours after it shuts down, the behavior is similar to the server startup activity.
Rejoining cluster after shutdown for an extended period	Perform a hot backup on the host running the master database before restarting a server that has been down for a long time. This avoids the potentially high cost synchronizing the server with the master during startup.

Two-Member Clustering

In this strategy the NNC cluster contains only two members in the cluster as shown in the diagram below.



As illustrated in the diagram above, the cluster in this case consists of only two host machines. Considering for the moment only the database replication mechanism the following points need to be appreciated:

- Most of the items described for a three member cluster pertain here as well.

- The most important difference is that with two members the quorum policy cannot be adopted since if a network partition exists between the host machines transactions would consistently fail.
- In order to ensure transactions are serviced if a network partition exists both member will act as master databases. When the networks is re-established the members will elect which one will shut down, leaving the other to maintain services.

HA for Two Member Cluster

In a two member cluster, high availability is provided for as follows:

- Database replication group:
 - Maintains one master in the cluster with one replica.
 - On master database failure, the remaining replica becomes master.
 - If a network partition exists both members become masters. On re-establishing network connectivity one of the member will shutdown.
- Messaging event and data distribution:
 - Remains the same as described for multi member cluster
- On-demand device configuration loading:
 - Remains the same as described for multi member cluster

Database Group Replication HA Behavior in a Two Member Cluster

The following table lists the activities involved in database replication HA behavior in a two member cluster setup.

Server startup	When an NNC server is started it joins the cluster as a replica and election is held if there is currently no master. If the member ends up as a replica, then it is synchronized with the master during the initialization phase of the database service startup.
Master member failure	When the master fails the remaining replica becomes the new master.
Transactions (Quorum)	Transactions return successfully if a majority of the members in the cluster have replied that they received the replicated datasets. If quorum in replies from replicas is not achieved in a specific time period, the transaction fails.
Network partition	When the master fails the remaining replica becomes the new master.
Elections	An election can be won with a single vote. This allows the replica to be elected master in the case the master fails.

Recovery after a network partition	In a two node cluster it's possible for the network connection between the master and replica to be partitioned or become unresponsive due to network latency. In this situation an election is held and both nodes are elected and act as masters. While in this state, write transactions can occur at both sites. As a result, special handling is required after the partition is resolved and the system recovers from a two master configuration to a single master configuration: • Before the partition is resolved both nodes will be in the role of master. • After the partition is resolved an election is automatically held to elect a master. • When the election is complete the node that wins remains the master and the other will become the replica. • The node that loses the election and becomes the replica after previously operating as master is immediately shut down. The node that loses the election after a partition is resolved is shut down because the two nodes might not be in synch. All Write transactions that occurred while the nodes were partitioned cannot be automatically resolved. Only Write transactions that occurred on the node that is elected master after the partition are accepted. Because Write transactions might have also occurred on the other node, it is shutdown and must be manually synchronized with the master before it is restarted.
Rejoining a cluster after graceful shutdown	An election can be won with a single vote. This allows the replica to be elected master in the case the master fails.
Rejoining cluster after shutdown for extended period	Perform a hot backup on the host running the master database before restarting a server that has been down for a long time. This avoids the potentially high cost synchronizing the server with the master during startup.

Data Synchronization Scenarios

The following sections describes different data synchronization scenarios.

Adding Member to a Cluster

You add a member to an existing cluster using the following general steps:. See *The Net-Net Central Installation Guide* for details.

Note: Adding a new member to a cluster or creating a new cluster during planned downtime.

1. Install Net-Net Central on the server you plan to add to the cluster.
2. Run the setup.sh script and add all previous members in a cluster to this new member's neighbor list.
3. Log into the running cluster bring up the HM console and record which server is running the master database.
4. Stop the replica members first, followed by the master.
5. Backup the databases on the host where the master database was running.

Note: If the member being added to the cluster contained a database at any point in time, run the bin/reinitialize.sh script.

6. Restore the database on the new member that is joining the cluster.
7. Run the setup.sh script on all other members of the cluster and add the new member to their neighbor list.
8. Start the master server first, and then the replica servers.

Member of a Cluster Fails or is Shutdown

The following describes automatic sequences that take place upon member failure or shutdown.

- Berkley DB XML: if the member leaving is running the master database, the remaining replicas carry out an election and elect a new master.
- MOM: maintain topic messages for any durable subscribers registered on the host that left. The messages would be maintained for a 24 hour period before truncation clean up occurred.
- Services that share common task processing or locks ensure that any locks acquired by the services on the member that left the cluster are cleaned up. Submitted tasks will still be processed by the remaining cluster members. Examples of these tasks include:
 - Save and Activate
 - Poller
- Clean up of any locks maintained for device synchronization. The service acquiring a lock generally provides an age-out time. If the original service cannot clean up the lock, it automatically gets aged out.
- Tasks initiated by the member that left the cluster would be re-submitted,
- Load balancers on the active hosts in the cluster bypass the front end server that has left the cluster. Clients are redirected to a valid running host.

- Health monitoring service: Health heartbeat monitor determining that a member's heartbeat has failed publishes a failed membership message to the cluster to inform all registered subscribers that a member has left the cluster.

Member Rejoining a Cluster

When a member that was previously part of a cluster rejoins that cluster, data resynchronization is required for the following databases:

- DB XML local master
- DB XML belonging to a replication group.

Types of members rejoining a cluster include those effected by the following:

- Temporary communication outage: All cluster members are running but network issues temporarily cause members to drop connections to each other.

The tolerance window for communication outage is 60 seconds.

Because of dynamic reconnection features, a member can temporarily leave the cluster for any reason. As long as the member NNC server has not been out of contact from the cluster for too long rejoining and re-synchronization is automatic. The Berkley DB XML will carry out re-elections and replicate datasets to replicas. The MOM brokers send messages missed by other brokers.

- NNC member fails or is stopped. A server fails outright and is down for some time until the administrator executes the startnnc.sh script. Determining how long the member has been out of contact is important:
 - If the member does not rejoin during the 24 hours of message storage, missed messages might not be sent to the joining broker. This server should be considered out of the cluster for extended period of time and needs to be treated as a member that needs to be re-synchronized before re-entering the cluster. If this is the case the following procedure is needed to re-join a member that has been out of the cluster for an extended period of time.
 1. Log into the running cluster, bring up the HM console and record which server is running the master database.
 2. Backup the databases on the host where the master database was running.

Note: If the member being added to the cluster contained a database at any point in time, run the bin/reinitialize.sh script.

3. Restore the database on the member that is re-joining the cluster.
 4. Start all NNC servers in the cluster that includes the new member
- The existing master database automatically detects a replica trying to rejoin and performs an internal initialization. Because internal initialization requires transfer of log file records that were missed to the client, it can take a lengthier period of time and could require database handles to be reopened in the replica's applications.

High Availability Scenarios

The following instructions are for different high availability scenarios. They contain a summary of steps, refer to the *Net-Net Central Installation Guide* for details.

Setup a Cluster

To setup a cluster for the first time:

1. Run setup.sh on each server in the cluster.
2. Choose the Custom setup option.
3. Apply the new license.
4. Configure HTTP/HTTPS.
5. Configure sudo password.

Entering Member Information

1. Choose Net-Net Central cluster management option.
2. Choose Configure and manage members in cluster.
3. Enter Yes to continue.
4. Choose Add a new member.
5. Enter Yes to continue.
6. Enter the member IP address.
7. Repeat steps 4 through 6 to add additional members to the cluster.
8. Choose Apply new cluster configuration when done entering new members to the cluster.
9. Enter Yes to continue.
10. Choose Quit out of cluster configuration.
11. Enter Yes to continue.

Configuring SFTP Properties

1. Choose Yes when prompted whether the machine will be a member of a cluster.
2. Enter Yes to continue.
3. Enter the username used to SFTP files off the machine.
4. Enter the password for the username.
5. Choose Quit setup to finish.

Start the Servers

Start the cluster members at the same time, or one at a time. The first server started becomes the master server for the cluster.

Shutdown a Cluster

To shutdown a cluster:

In order to avoid unnecessary database elections, first shutdown the servers running the replica databases, and then shutdown the server running the master.

1. Login to the NNC GUI client and access the Health Monitor to identify the server with the master database.
2. Identify the master database in one of two ways:
 - In the Heartbeat console, the server that is running the master database has (master) next to the IP address.
 - Search the logs/DbService.log file for a line that looks like this:


```
2011-06-14 16:52:16,266 INFO
[com.acmepacket.ems.server.services.database.ReplicatedXMLDatabaseManagerImpl] - Method: [setCurrentState] Thread: [Thread-16:74]
Msg:[Current State = unknown, New State = master]
2011-06-14 16:52:16,295 INFO
[com.acmepacket.ems.server.services.database.ReplicatedXMLDatabaseManagerImpl] - Method: [initDbEnv] Thread: [DatabaseService:20]
Msg:[Replicated Database environment initialization complete. Role = master,
IPAddress = 172.30.80.19]
```
3. Login to the other servers that are running replica databases first and run the shutdownncc.sh script on each.
4. Once these servers have shutdown, login to the server running the master database and run the shutdownnnc.sh script.

Retiring Cluster Member

To retire a member from a cluster during planned downtime:

1. Shutdown the node to be removed from the cluster.
2. Uninstall Net-Net Central from that node.
3. Shutdown all remaining nodes starting with the replicas, followed by the master.
4. Run setup.sh
5. Choose the Custom setup option.
6. Choose Net-Net Central cluster management.
7. Enter Yes to continue.
8. Choose Configure and manage members in a cluster.
9. Enter Yes to continue.
10. Choose Remove all remote members to remove the first node from the cluster configuration.
11. Enter Yes to continue.
12. Choose Proceed with removing all remote members.
13. One by one, add all members to the cluster, excluding the retired member.
14. Enter Yes to continue.
15. Remove all of the files from the db/replicated directory except for DB_CONFIG.

16. Startup the master node, followed by the replicas.

After startup is complete, ensure the replications between the two nodes is in synch.

Repeat steps for each remaining node in the cluster.

Network Partition in Two Node Cluster

You are running a two node cluster when connectivity between the two nodes is lost. Each node elects itself as master. At this point, it is possible to lose any writes that occur to one of the databases. When the network connectivity is re-established automatically, the node that loses the election to be the master shuts down. The writes that occurred to that node are lost.

1. To re-establish the cluster, perform a hot backup on the node that is still operational.
2. Run the bin/reinitialize.sh script on the shutdown node.
3. Restore the database backup from the master onto the shutdown node.
4. Startup the shutdown node.

Network Partition in a Three Node Cluster

You are running a three or more node cluster. The network connectivity between two or more of the nodes is lost. The partition that contains a majority of the nodes elects a master from among those nodes. Any partition containing a number of servers less than the majority transitions to a READ-ONLY mode. Database updates are not permitted until the partition is resolved or the cluster has been reconfigured. A four node cluster that is partitioned into two 2 node clusters will contain two clusters running in READ-ONLY mode. When network connectivity is re-established, one node is elected master and the cluster resumes normal operation.

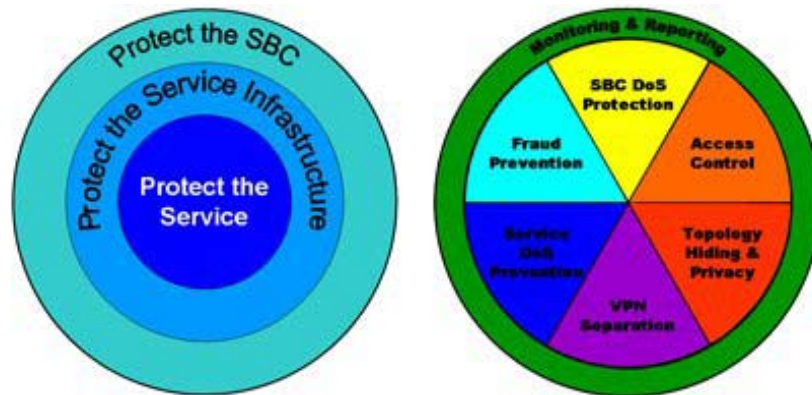
Introduction

This chapter explains Net-Net SBC security, which is designed to provide security for VoIP and other multimedia services. It includes access control, DoS attack, and overload protection, which helps secure service and protect the network infrastructure (including the Net-Net SBC). In addition, Net-Net SBC security lets legitimate users still place calls during attack conditions; protecting the service itself.

Security Overview

Net-Net SBC security includes the Net-SAFE framework's numerous features and architecture designs. Net-SAFE is a requirements framework for the components required to provide protection for the Session Border Controller (SBC), the service provider's infrastructure equipment (proxies, gateways, call agents, application servers, and so on), and the service itself.

The following diagrams illustrate Net-SAFE:



Each of Net-SAFE's seven functions consists of a collection of more specific features:

- Session border controller DoS protection: autonomic, SBC self-protection against malicious and non-malicious DoS attacks and overloads at Layers 2 to 4 (TCP, SYN, ICMP, fragments, and so on) and Layers 5 to 7 (SIP signaling floods, malformed messages, and so on).
- Access control: session-aware access control for signaling and media using static and dynamic permit/deny access control lists (ACLs) at layer 3 and 5.
- Topology hiding and privacy: complete infrastructure topology hiding at all protocol layers for confidentiality and attack prevention security. Also, modification, removal or insertion of call signaling application headers and fields. Includes support for the SIP Privacy RFC.

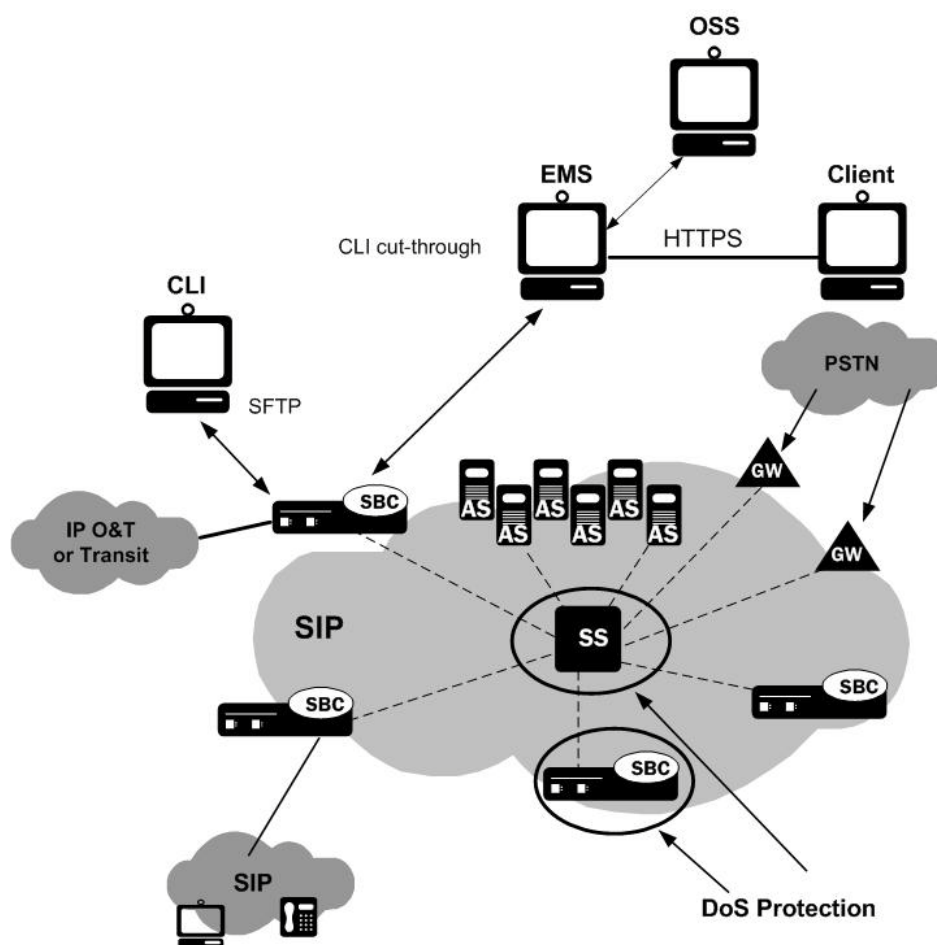
- VPN separation: support for Virtual Private Networks (VPNs) with full inter-VPN topology hiding and separation, ability to create separate signaling and media-only VPNs, and with optional intra-VPN media hair-pinning to monitor calls within a VPN.
- Service infrastructure DoS prevention: per-device signaling and media overload control, with deep packet inspection and call rate control to prevent DoS attacks from reaching service infrastructure such as SIP servers, softswitches, application servers, media servers or media gateways.
- Fraud prevention: session-based authentication, authorization, and contract enforcement for signaling and media; and service theft protection.
- Monitoring and reporting: audit trails, event logs, access violation logs and traps, management access command recording, Call Detail Records (CDRs) with media performance monitoring, raw packet capture ability and lawful intercept capability. The monitoring method itself is also secured, through the use of SSH and SFTP, and through the ability to use a separate physical Ethernet port for management access.

Denial of Service Protection

This section explains the Denial of Service (DoS) protection for the Net-Net SBC. The Net-Net SBC DoS protection functionality protects softswitches and gateways with overload protection, dynamic and static access control, and trusted device classification and separation at Layers 3-5. The Net-Net SBC itself is protected from signaling and media overload, but more importantly the feature allows legitimate, trusted devices to continue receiving service even during an attack. DoS protection prevents the Net-Net SBC host processor from being overwhelmed by a targeted DoS attack from the following:

- IP packets from an untrusted source as defined by provisioned or dynamic ACLs
- IP packets for unsupported or disabled protocols
- Nonconforming/malformed (garbage) packets to signaling ports
- Volume-based attack (flood) of valid or invalid call requests, signaling messages, and so on.
- Overload of valid or invalid call requests from legitimate, trusted sources

The following diagram illustrates DoS protection applied to the softswitch and to the Net-Net SBC.



Levels of DoS Protection

The multi-level Net-Net SBC DoS protection consists of the following strategies:

- **Fast path filtering/access control**: access control for signaling packets destined for the Net-Net SBC host processor as well as media (RTP) packets. The Net-Net SBC performs media filtering by using the existing dynamic pinhole firewall capabilities. Fast path filtering packets destined for the host processor require the configuration and management of a trusted list and a deny list for each Net-Net SBC realm (although the actual devices can be dynamically trusted or denied by the Net-Net SBC based on configuration). You do not have to provision every endpoint/device on the Net-Net SBC, but instead retain the default values.
- **Host path protection**: includes flow classification, host path policing and unique signaling flow policing. Fast path filtering alone cannot protect the Net-Net SBC host processor from being overwhelmed by a malicious attack from a trusted source. The host path and individual signaling flows must be policed to ensure that a volume-based attack will not overwhelm the Net-Net SBC's normal call processing; and subsequently not overwhelm systems beyond it.

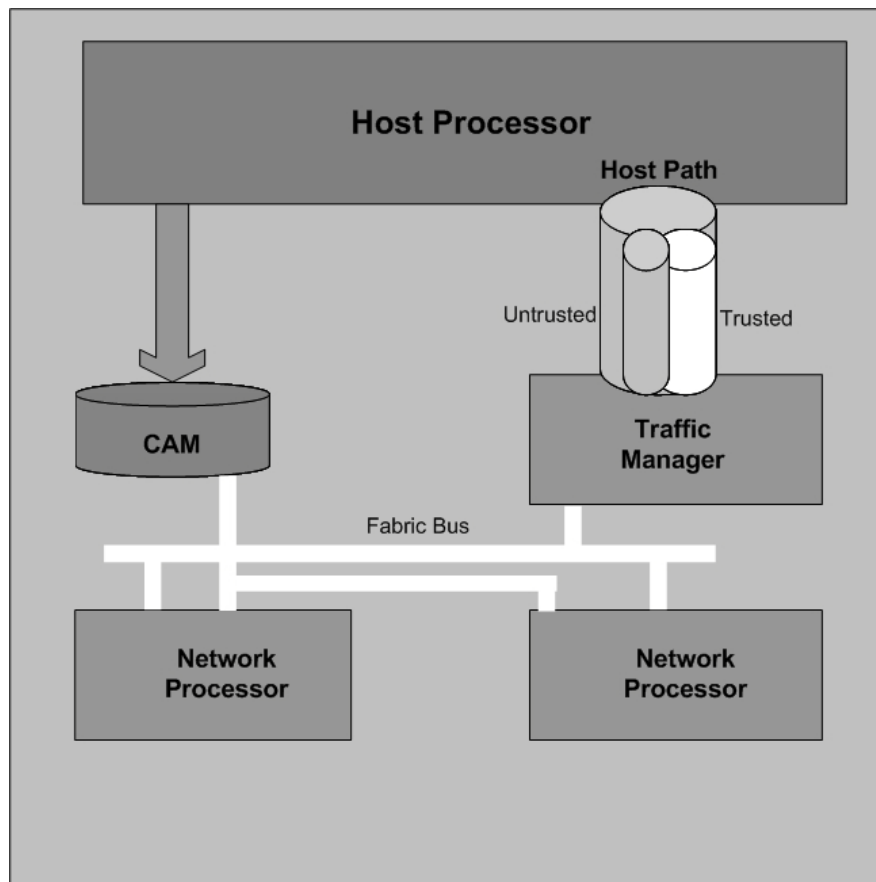
The Net-Net SBC must classify each source based on its ability to pass certain criteria that is signaling- and application-dependent. At first each source is considered untrusted with the possibility of being promoted to fully trusted. The Net-Net SBC maintains two host paths, one for each class of traffic (trusted and untrusted), with different policing characteristics to ensure that fully trusted traffic always gets precedence.

- Host-based malicious source detection and isolation – dynamic deny list. Malicious sources can be automatically detected in real-time and denied in the fast path to block them from reaching the host processor.

About the Process

DoS attacks are handled in the Net-Net SBC's host path. The Net-Net SBC uses NAT table entries to filter out undesirable IP addresses; creating a deny list. After a packet from an endpoint is accepted through NAT filtering, policing is implemented in the Traffic Manager subsystem based on the sender's IP address. NAT table entries distinguish signaling packets coming in from different sources for policing purposes. The maximum number of policed calls that the Net-Net SBC can support is 16K (on 32K CAM / IDT CAM).

The Traffic Manager has two pipes, trusted and untrusted, for the signaling path. Each signaling packet destined for the host CPU traverses one of these two pipes.



Trusted Path

Packets from trusted devices travel through the trusted pipe in their own individual queues. In the Trusted path, each trusted device flow has its own individual queue (or pipe). The Net-Net SBC can dynamically add device flows to the trusted list by promoting them from the Untrusted path based on behavior; or they can be statically provisioned.

Trusted traffic is put into its own queue and defined as a device flow based on the following:

- source IP address
- source UDP/TCP port number
- destination IP address
- destination UDP/TCP port (SIP or MGCP interface to which it is sending)
- realm it belongs to, which inherits the Ethernet interface and VLAN it came in on

For example, SIP packets coming from 10.1.2.3 with UDP port 1234 to the Net-Net SBC SIP interface address 11.9.8.7 port 5060, on VLAN 3 of Ethernet interface 0:1, are in a separate Trusted queue and policed independently from SIP packets coming from 10.1.2.3 with UDP port 3456 to the same Net-Net SBC address, port and interface.

Data in this flow is policed according to the configured parameters for the specific device flow, if statically provisioned. Alternatively, the realm to which endpoints belong have a default policing value that every device flow will use. The defaults configured in the realm mean each device flow gets its own queue using the policing values. As shown in the previous example, if both device flows are from the same realm and the realm is configured to have an average rate limit of 10K bytes per second (10KBps), each device flow will have its own 10KBps queue. They are not aggregated into a 10KBps queue.

The individual flow queues and policing lets the Net-Net SBC provide each trusted device its own share of the signaling, separate the device's traffic from other trusted and untrusted traffic, and police its traffic so that it can't attack or overload the Net-Net SBC (therefore it is trusted, but not completely).

Address Resolution Protocol Flow

The Address Resolution Protocol (ARP) packets are given their own trusted flow with the bandwidth limitation of 8 Kbps. ARP packets are able to flow smoothly, even when a DoS attack is occurring.

Untrusted Path

Packets (fragmented and unfragmented) that are not part of the trusted or denied list travel through the untrusted pipe. In the untrusted path, traffic from each user/device goes into one of 2048 queues with other untrusted traffic. Packets from a single device flow always use the same queue of the 2048 untrusted queues, and 1/2048th of the untrusted population also uses that same queue. To prevent one untrusted endpoint from using all the pipe's bandwidth, the 2048 flows defined within the path are scheduled in a fair-access method. As soon as the Net-Net SBC decides the device flow is legitimate, it will promote it to its own trusted queue.

All 2048 untrusted queues have dynamic sizing ability, which allows one untrusted queue to grow in size, as long as other untrusted queues are not being used proportionally as much. This dynamic queue sizing allows one queue to use more than average when it is available. For example, in the case where one device flow represents a PBX or some other larger volume device. If the overall amount of untrusted packets grows too large, the queue sizes rebalance, so that a flood attack or DoS attack does not create excessive delay for other untrusted devices.

In the usual attack situations, the signaling processor detects the attack and dynamically demotes the device to denied in the hardware by adding it to the deny ACL list. Even if the Net-Net SBC does not detect an attack, the untrusted path gets serviced by the signaling processor in a fair access mechanism. An attack by an untrusted device will only impact 1/1000th of the overall population of untrusted devices, in the worst case. Even then there's a probability of users in the same 1/1000th percentile getting in and getting promoted to trusted.

IP Fragment Packet Flow

All fragment packets are sent through their own 1024 untrusted flows in the Traffic Manager. The first ten bits (LSB) of the source address are used to determine which fragment-flow the packet belongs to. These 1024 fragment flows share untrusted bandwidth with already existing untrusted-flows. In total, there are 2049 untrusted flows: 1024-non-fragment flows, 1024 fragment flows, and 1 control flow.

Fragmented ICMP packets are qualified as ICMP packets rather than fragment packets. Fragment and non-fragmented ICMP packets follow the trusted-ICMP-flow in the Traffic Manager, with a bandwidth limit of 8Kbs.

Static and Dynamic ACL Entry Limits

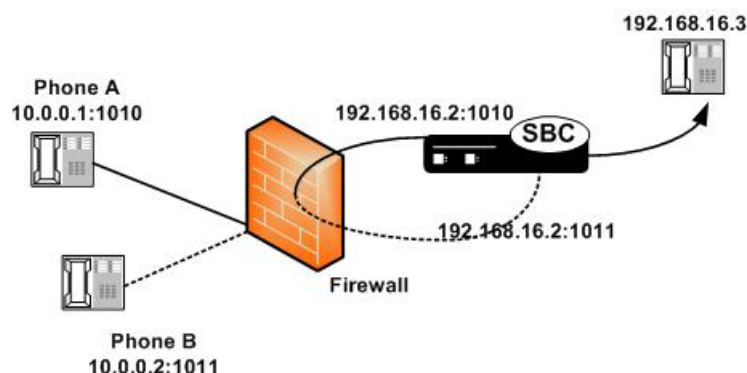
The Net-Net SBC can simultaneously police a maximum of 16,000 trusted device flows, while at the same time denying an additional 16,000 attackers. If all 16,000 trusted entries and 16,000 denied entries are being used, the Net-Net SBC can handle at most 32,000 simultaneous media flows (which is 16,000 simultaneous calls for normal voice calls). The usage is dynamic, so that when the Net-Net SBC uses fewer trusted or denied entries, additional capacity for call media handling is freed. These limits let the Net-Net SBC handle at least 32,000 simultaneous media flows, with the potential of up to 64,000. If list space becomes full and additional device flows need to be added, the oldest entries in the list are removed and the new device flows are added.

Dynamic Deny for HNT

Dynamic deny for HNT has been implemented on the Net-Net SBC for cases when callers are behind a NAT or firewall. Without this feature, if one caller behind a NAT or firewall were denied, the Net-Net SBC would also deny all other users behind the same NAT or firewall. This would be true even for endpoints behind the firewall that had not crossed threshold limits you set for their realm; all endpoints behind the firewall would go out of service. In the following diagram, both Phone A and Phone B would be denied because their IP addresses would be translated by the firewall to the same IPv4 address (192.168.16.2).

However, dynamic deny for HNT allows the Net-Net SBC to determine, based on the UDP/TCP port, which endpoints should be denied and which should be allowed. The Net-Net SBC can determine that even though multiple endpoints originating behind a firewall appear with the same IPv4 address, those addresses use different ports and are unique.

As shown in the diagram below, the ports from Phone A and Phone B remain unchanged. This way, if Phone A violates the thresholds you have configured, the Net-Net SBC can block traffic from Phone A while still accepting traffic from Phone B.



Host and Media Path Protection Process

The Net-Net SBC Network Processors (NPs) check the deny and permit lists for received packets, and classify them as trusted, untrusted or denied (discard). Only packets to signaling ports and dynamically signaled media ports are permitted. All other packets sent to Net-Net SBC ports are filtered. Only packets from trusted and untrusted (unknown) sources are permitted; any packet from a denied source is dropped by the NP hardware. The Traffic Manager manages bandwidth policing for trusted and untrusted traffic, as described earlier. Malicious traffic is detected in the host processor and the offending device is dynamically added to denied list, which enables early discard by the NP. Devices become trusted based on behavior detected by the Signaling Processor, and dynamically added to the trusted list. This process enables the proper classification by the NP hardware. All other traffic is untrusted (unknown).

Session Border Controller Access Control

You can create static trusted/untrusted/deny lists with source IP addresses or IP address prefixes, UDP/TCP port number or ranges, and based on the appropriate signaling protocols. Furthermore, the Net-Net SBC can dynamically promote and demote device flows based on the behavior, and thus dynamically creates trusted, untrusted, and denied list entries.

Access Control for Hosts

ACLs are supported for all VoIP signaling protocols on the Net-Net SBC: SIP, H.323, and MGCP. The Net-Net SBC loads ACLs so they are applied when signaling ports are loaded. The following rules apply to static NAT entries based on your configuration:

- If there are no ACLs applied to a realm that have the same configured trust level as that realm, the Net-Net SBC adds a default NAT entry using the realm parameters.
- If you configure a realm with **none** as its trust level and you have configured ACLs, the Net-Net SBC only applies the ACLs.
- If you set a trust level for the ACL that is lower than the one you set for the realm, the Net-Net SBC will not add a separate NAT entry for the ACL.

ACLs provide access control based on destination addresses when you configure destination addresses as a way to filter traffic. You can set up a list of access control exceptions based on the source or the destination of the traffic.

For dynamic ACLs based on the promotion and demotion of endpoints, the rules of the matching ACL are applied.

Media Access Control

The media access control consists of media path protection and pinholes through the firewall. Only RTP and RTCP packets from ports dynamically negotiated through signaling (SIP, H.323, MGCP) are allowed, which reduces the chance of RTP hijacking. Media access depends on both the destination and source RTP/RTCP UDP port numbers being correct, for both sides of the call.

Host Path Traffic Management

The host path traffic management consists of the dual host paths discussed earlier:

- Trusted path is for traffic classified by the Net-Net SBC as trusted. You can initially define trusted traffic by ACLs, as well as by dynamically promoting it through successful SIP or MGCP registration, or a successful call establishment. You can configure specific policing parameters per ACL, as well as define default policing values for dynamically-classified flows. Traffic for each trusted device flow is limited from exceeding the configured values in hardware. Even an attack from a trusted, or spoofed trusted, device cannot impact the system.
- Untrusted path is the default for all unknown traffic that has not been statically provisioned otherwise. For example, traffic from unregistered endpoints. Pre-configured bandwidth policing for all hosts in the untrusted path occurs on a per-queue and aggregate basis.

Traffic Promotion

Traffic is promoted from untrusted to trusted list when the following occurs:

- successful SIP registration for SIP endpoints
- successful RSIP response for MGCP endpoints
- successful session establishment for SIP or MGCP calls

Malicious Source Blocking

Malicious source blocking consists of monitoring the following metrics for each source:

- SIP transaction rate (messages per second)
- SIP call rate (call attempts per second)
- Nonconformance/invalid signaling packet rate

Device flows that exceed the configured invalid signaling threshold, or the configured valid signaling threshold, within the configured time period are demoted, either from trusted to untrusted, or from untrusted to denied classification.

Blocking Actions

Blocking actions include the following:

- Dynamic deny entry added, which can be viewed through the ACLI.
- SNMP trap generated, identifying the malicious source

Dynamically added deny entries expire and are promoted back to untrusted after a configured default deny period time. You can also manually clear a dynamically added entry from the denied list using the ACLI.

Protecting Against Session Agent Overloads

You can prevent session agent overloads with registrations by specifying the registrations per second that can be sent to a session agent.

DoS Flow Classification Allocation

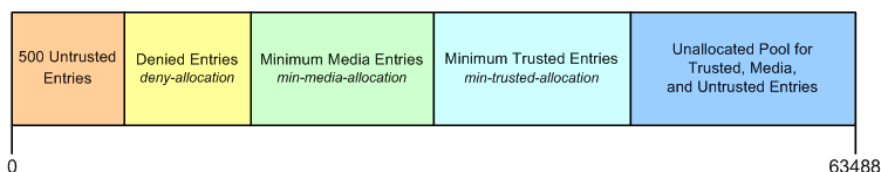
The Net-Net SBC lets you set limits for three of the four types of flow classification used for DoS functionality. In previous releases, the Net-Net SBC assigned 14,000 flows for denied entries, 16,000 flows for trusted entries, ~32,000 flows for media entries, and ~1000 for untrusted entries.

The DoS flow classification allocation feature allows you to configure a minimum number of media entries, a minimum number of trusted entries, and a set number of denied entries.

Apportioning Flow Types

The total number flow classification entries is 63488. The number of untrusted entries is set to 500. The number of denied flows, once configured, remains static. This number can range from 0 to 62988 (63488 - 500 used for untrusted). The Net-Net SBC cannot accept more denied flows than the number you configure.

Media and trusted flows are assigned dynamically, within certain constraints. You configure a minimum number of flows that the Net-Net SBC reserves for these two flow types individually. The values of the minimum media and minimum trusted flows can each be as great as the total number of flows available to flow classification (63488) - 500 used for untrusted flows - the configured number of denied flows. This value represents the total number of flows you have left to assign to both media and trusted flows.



After all entries are used up from either the Minimum Media or Minimum Trusted entries, the Net-Net SBC takes flow entries from the unallocated pool for either of the two classes. In this way, DoS entries are dynamically assigned. When there is contention between allocating a flow to either a media or a trusted flow, the Net-Net SBC gives the flow entry to the media flow.

Configuration Overview

Configuring Net-Net SBC DoS protection includes masking source IP and port parameters to include more than one match and configuring guaranteed minimum bandwidth for trusted and untrusted signaling path. You can also configure signaling path policing parameters for individual source addresses. Policing parameters are defined as peak data rate (in bytes/sec), average data rate (in bytes/sec), and maximum burst size.

You can configure deny list rules based on the following:

- ingress realm
- source IP address
- source port
- transport protocol (TCP/UDP)
- application protocol (SIP, MGCP, H.323)

Changing the Default Net-Net SBC Behavior

The Net-Net SBC automatically creates permit untrusted ACLs that let all sources (address prefix of 0.0.0.0/0) reach each configured realm's signaling interfaces, regardless of the realm's address prefix. To deny sources or classify them as trusted, you create static or dynamic ACLs, and the global permit untrusted ACL to specifically deny sources or classify them as trusted. Doing this creates a default permit-all policy with specific deny and permit ACLs based on the realm address prefix.

You can change that behavior by configuring static ACLs for realms with the same source prefix as the realm's address prefix; and with the trust level set to the same value as the realm. Doing this prevents the permit untrusted ACLs from being installed. You then have a default deny all ACL policy with specific static permit ACLs to allow packets into the system.

Example 1: Limiting Access to a Specific Address Prefix Range

The following example shows how to install a permit untrusted ACL of source 12.34.0.0/16 for each signalling interface/port of a realm called `access`. Only packets from within the source address prefix range 12.34.0.0/16, destined for the signaling interfaces/port of the realm named `access`, are allowed. The packets go into untrusted queues until they are dynamically demoted or promoted based on their behavior. All other packets are denied/dropped.

- Configure a realm called `access` and set the trust level to low and the address prefix to 12.34.0.0/16.
- Configure a static ACL with a source prefix of 12.34.0.0/16 with the trust level set to low for the realm named `access`.

Example 2: Classifying the Packets as Trusted

Building on Example 1, this example shows how to classify all packets from 12.34.0.0/16 to the realm signaling interfaces as trusted and place them in a trusted queue. All other packets from outside the prefix range destined to the realm's signaling interfaces are allowed and classified as untrusted; then promoted or demoted based on behavior.

You do this by adding a global permit untrusted ACL (source 0.0.0.0) for each signaling interface/port of the **access** realm. You configure a static ACL with a source prefix 12.34.0.0/16 and set the trust level to high.

Adding this ACL causes the Net-Net SBC to also add a permit trusted ACL with a source prefix of 12.34.0.0/16 for each signaling interface/port of the **access** realm. This ACL is added because the trust level of the ACL you just added is high and the realm's trust level is set to low. The trust levels must match to remove the global permit trusted ACL.

Example 3: Installing Only Static ACLs

This example shows you how to prevent the Net-Net SBC from installing the global permit (0.0.0.0) untrusted ACL.

- Configure a realm with a trust level of **none**.
- Configure static ACLs for that realm with the same source address prefix as the realm's address prefix, and set the trust level to any value.

The Net-Net system installs only the static ACLs you configure.

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by admin@console

Last modified date 2011-04-26 18:27:37

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net Central.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net Central identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net Central is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

EMS_admin_172.30.80.2_UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net Central, the date timestamp is specified in UTC.

If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Dynamic Demotion for NAT Devices

In addition to the various ways the Net-Net SBC already allows you to promote and demote devices to protect itself and other network elements from DoS attacks, it can now block off an entire NAT device. The Net-Net SBC can detect when a configurable number of devices behind a NAT have been blocked off, and then shut off the entire NAT's access.

This dynamic demotion of NAT devices can be enabled for an access control (ACL) configuration or for a realm configuration. When you enable the feature, the Net-Net SBC tracks the number of endpoints behind a single NAT that have been labeled untrusted. It shuts off the NAT's access when the number reaches the limit you set.

The demoted NAT device then remains on the untrusted list for the length of the time you set in the **deny-period**.

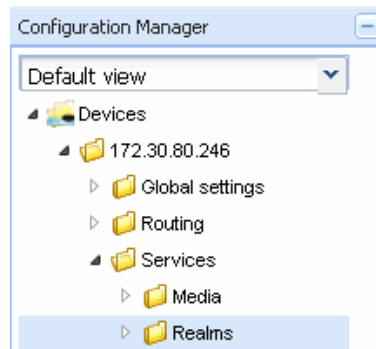
Enabling Dynamic Demotion of NAT Devices

You can enable this feature either in the ACL configuration, or on a per-realm basis.

Dynamic NAT Device Demotion for ACLs

To enable dynamic demotion of NAT devices for an ACL:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the realm table and click **Edit**.
4. Click the Access control tab at the top of the content area.
5. Click a realm in the Access control table and click **Edit**.
6. **NAT trust threshold (endpoints per device)**—Enter the maximum acceptable number of untrusted endpoints allowed before the Net-Net SBC demotes the entire NAT device to untrusted. The default is 0, meaning dynamic demotion of NAT devices is disabled.

NAT trust threshold (endpoints per device)

7. Click **Apply**.

Configuring DoS Security

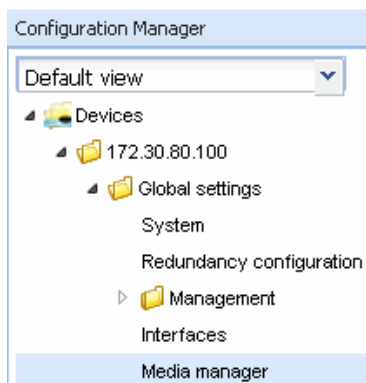
This section describes how to configure security settings on your Net-Net SBC from the Net-Net Central. Configuring security in Net-Net Central is handled in a different order than in the ACLI. The order of configuration in Net-Net Central is:

- Media manager
- Realm and access control per realm

Host Access Policing

To configure host access policing:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Click the CPU Policing and DoS tab at the top of the content area.
5. **Total signaling bandwidth reserved for host-path(bytes/sec)**—Enter the maximum bandwidth available for this host path in bytes per second. This value also includes any Telnet and FTP traffic on media ports.
 - minimum: 71,000
 - maximum: 10,000,000 (default)
6. **Percentage bandwidth reserved for untrusted sources(%)**—Enter the percentage of signaling bandwidth guaranteed for untrusted hosts. The rest of the bandwidth is available for trusted resources, but can also be used for untrusted sources (see max-untrusted-signaling).
 - minimum: 1%
 - maximum: 100%

The default value is 30%.
7. **Percentage of unused bandwidth available for untrusted sources(%)**—Enter the percentage of signaling bandwidth that can be used by untrusted hosts. This bandwidth is only available when not being used by trusted sources.
 - minimum: 1%
 - maximum: 100% (default)
8. **Tolerance window size for access limit(sec)**—Enter the tolerance-window size in seconds to measure host access limits. The value entered here is used to

measure the invalid message rate and maximum message rate for the realm configuration.

Signaling CPU policing

Total signaling bandwidth reserved for host-path (bytes per second)	10000000
Percentage bandwidth reserved for untrusted sources(%)	30
Percentage of unused bandwidth available for untrusted sources(%)	100
Tolerance window size for access limit(sec)	30

- Click **Apply**.

Basic Access Control for a Realm

Each host within a realm can be policed based on average rate, peak rate, and maximum burst size of signaling messages. These parameters take effect only when the host is trusted. You can also set the trust level for the host within the realm. All untrusted hosts share the bandwidth defined for the media manager: maximum untrusted bandwidth and minimum untrusted bandwidth.

Adding Access Control

- Click **Add** next to the Access control exceptions area. The Add Access control dialog box appears.
- Scroll to Source address.
- Port**—Enter the source port for this address.
- IP Address**—Enter the source IPv4 address for the host.
- Number of port bits**—Enter the number of port bits to be matched to this source. You do not need to specify the port bits if you want the exact port to be matched.
- Number of bits**—Enter the number of address bits to be matched to this source. You do not need to specify the number of address bits if you want all 32 bits of the address to be matched.
- Scroll to Destination Address.
- Port**—Enter the destination port. Not specifying a port implies an exact source port.
- IP Address**—Enter the destination IPv4 address.
- Number of port bits**—Enter the number of port bits for the destination port.
- Number of bits**—Enter the number of address bits for the destination address.
- Transmission protocol**—Select the transport protocol for this ACL from the combination box.
 - ALL
 - TCP
 - UDP

13. **Application protocol**—Select the application protocol for this ACL from the combination box.
 - SIP
 - H.323
 - MGCP

Add Access control

*Realm: test (Unique required)

Source address

*Port: 0 (Range:0..65535, Default:0)

*IP Address: 110.54.32.10 (Enter valid IP address or prefix)

*Number of port bits: 0 (Range:0..4294967295, Default:0)

*Number of bits: 0 (Range:0..32, Default:0)

Destination address

*Port: 0 (Range:0..65535, Default:0)

*IP Address: 100.10.20.30 (Enter valid IP address or prefix)

*Number of port bits: 0 (Range:0..4294967295, Default:0)

*Number of bits: 0 (Range:0..32, Default:0)

*Application protocol: SIP (Unique required)

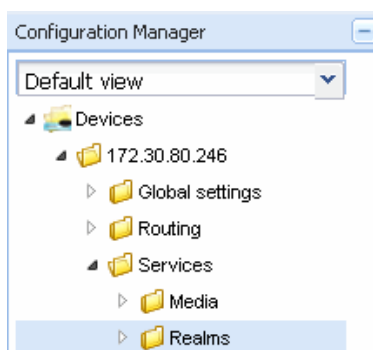
*Transmission protocol: TCP (Unique required, Default:ALL)

Apply Cancel

14. Click **Apply**. The entry appears in the Access control list.

Editing Access Control

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm from the table and click **Edit**.
4. Click the Access control tab at the top of the content area.
5. Select an entry in the Access control list and click **Edit**.
6. **Description**—Enter an optional descriptive string to identify the configuration and its function.

7. **Access**—Select the access control type for this ACL entry from the combination box.
 - permit: puts the entry into the untrusted list. The entry is promoted or demoted according to the trust level set for this host.
 - deny: puts the entry in the deny list.
8. **Sustained signaling data rate (bps)**—Enter the sustained rate in bytes per second for host path traffic from a trusted source within this realm.
 - minimum: zero (0) is disabled
 - maximum: 4294967295
9. **Trust level**—Select the trust level for the host in this realm from the combination box. The remaining textboxes area activated (unless you chose none.)
 - none: host is always untrusted. It is never promoted to the trusted list or demoted to the deny list.
 - low: host can be promoted to the trusted list or demoted to the deny list.
 - medium: host can be promoted to the trusted list but is only demoted to untrusted. It is never added to the deny list.
 - high: host is always trusted.
10. **NAT trust threshold**—Enter the maximum number of untrusted endpoints allowed before an entire NAT device is demoted to untrusted. 0 means dynamic demotion of NAT devices is disabled.
 - minimum: zero (0) is disabled
 - maximum: 65535
11. **CAC failure threshold**—Enter the number of CAC failures for any single endpoint that will demote it from the trusted queue to the untrusted queue for this realm.
 - minimum: zero (0) is disabled
 - maximum: $2^{32} - 1$
12. **Untrusted CAC failure threshold**—Enter the number of CAC failures for any single endpoint that will promote it from the untrusted queue to the trusted queue for this realm.
 - minimum: zero (0) is disabled
 - maximum: $2^{32} - 1$
13. **Maximum invalid signaling messages with tolerance window**—Enter the acceptable invalid signaling message rate within the tolerance window. The value you enter here is only valid when the trust level is low or medium.
 - minimum: zero (0) is disabled
 - maximum: 4294967295
14. **Maximum signaling messages with tolerance window**—Enter the acceptable signaling message rate within the tolerance window. The value you enter here is only valid when the trust level is low or medium.
 - minimum: zero (0) is disabled
 - maximum: 4294967295

15. **Messages from untrusted endpoints**—Enter the maximum number of untrusted messages the host can send within the tolerance window. Use to configure different values for trusted and un-trusted endpoints for valid signaling message parameters. Also configurable per realm.

- minimum: zero (0) is disabled
- maximum: 4294967295

The default value is 0.

16. **Deny period(s)**—Enter the time period in seconds after which the entry for this host is removed from the deny list.

- minimum: zero (0) seconds
- maximum: 4294967295 seconds

The default value is 30 seconds.

*Realm	test ▼
Description	
*Transmission protocol	ALL ▼
*Application protocol	SIP ▼
Access	permit ▼
Sustained signaling data rate (bps)	0
Trust level	none ▼
NAT trust threshold (endpoints per device)	0
CAC failure threshold	0
Untrusted CAC failure threshold	0
Maximum invalid signaling messages with tolerance window	0
Maximum signaling messages with tolerance window	0
Minimum reserved bandwidth (bps)	0
Messages from untrusted endpoints	0
Deny period(s)	30

17. Click **Apply** to continue.

18. Click **Apply**.

Managing RSIPs

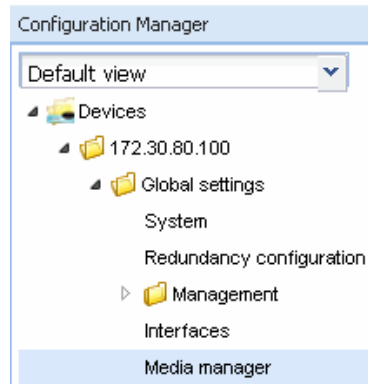
When multiple MGCP clients simultaneously reboot, RSIPs flood the Net-Net SBC through an untrusted pipe. AUEPs are also processed by the Net-Net SBC through its untrusted pipe. The AUEPs contend with the RSIPs flooding in for the available untrusted bandwidth. This contention can cause AUEPs to be lost, which in turn causes the MGCP client to retry the RSIPs; and a network/call outage can occur.

All received RSIPs are funneled through a 5700 Traffic Manager application pipe.

Configuring Managing RSIPs for DoS

To configure the RSIP traffic over the application pipe:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. **Signaling bandwidth guaranteed for app traffic**—Percentage of the untrusted bandwidth reserved for specific application messages. Currently the only supported application message is RSIP for MGCP and NCS.

Signaling bandwidth guaranteed for app traffic (bytes per second)

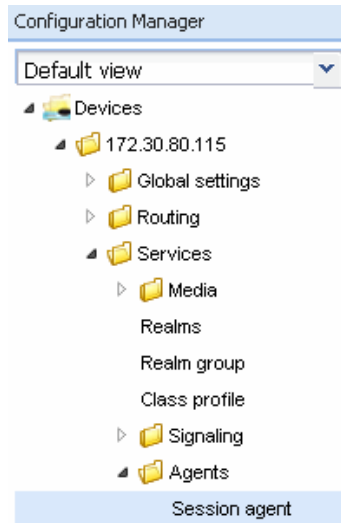
5. Click **Apply**.

Configuring Overload Protection for Session Agents

You can configure overload protection for session agents using the Net-Net Central.

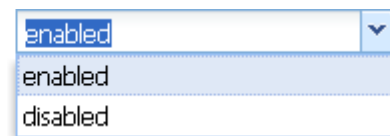
To configure overload protection for session agents

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the session agent you want to configure and click **Edit**.
4. Click the Constraints tab at the top of the content area.
5. **Session constraints**—Select enabled from the combination box to activate the constraint parameters.
6. **Max. inbound sessions (#)**—Enter a number to set the maximum number of inbound registrations per second you want sent to the session agent. Also configurable per realm.
 minimum: zero (0) seconds (default value)
 maximum: 4294967295 seconds
7. **Max. outbound sessions (#)**—Enter a number to set the maximum number of outbound registrations per second you want sent to the session agent. Also configurable per realm.
 minimum: zero (0) seconds (default value)
 maximum: 4294967295 seconds
8. **Sustained rate window (sec)**—Enter a number to set the sustained window period (in milliseconds) that is used to measure the sustained rate. (Refer to the max-sustain-rate information).
 - minimum: zero (0) milliseconds (default value)
 - maximum: 232 -1 milliseconds.

Session constraints



9. Click **Apply**.

Configuring DoS Flow Classification Allocation

Care must be given to how you assign flow classification entries for each of the three configurable parameters. You should constrain the media and trusted entries to values as large as possible to ensure that all calls are handled properly while

balancing that number with an adequate amount of denied entries to ensure that the Net-Net SBC is protected.

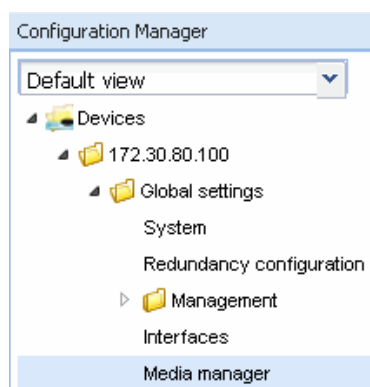
If you over allocate any of the three DoS Entry Allocation fields, you will not receive a warning message on the ACLI until after you activate the configuration. The same warning message will also appear in the log.npsoft logfile.

The following table lists the constraints of the DoS Flow Classification Allocation parameters:

Parameter	Range	Default
Denied Entries	0-62988	1000
Minimum Media Entries	0-62988	32000
Minimum Trusted Entries	0-62988	1000

To configure DoS flow classification allocation:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Click the CPU Policing and DoS tab at the top of the content area.
5. Scroll to DoS entry allocation.
6. **Minimum number of media entries supported**—Enter the minimum number of entries devoted specifically to media flows in kilobytes. The range is 0 to 245K.
7. **Minimum number of trusted entries supported**—Enter the minimum number of entries devoted specifically to trusted flows in kilobytes. The range is 0 to 128K.

DoS entry allocation

Minimum number of media entries supported	32000
Minimum number of trusted entries supported	1000

8. Click **Apply**.

Media Policing

Media policing controls the throughput of individual media flows in the Net-Net SBC, which in turn provides security and bandwidth management functionality. The media policing feature works for SIP, H.323, SIP-H.323, and MGCP/NCS protocols. The media policing feature also lets you police static flows and RTCP flows.

The term media policing refers to flows that go through the Net-Net SBC. Flows that are directed to the host application are not affected by media policing.

You can use media policing to protect against two potential security threats that can be directed against your Net-Net SBC:

- **Media DoS**—Once media flows are established through the Net-Net SBC, network resources are open to RTP media flooding. You can eliminate the threat

of a media DoS attack by constraining media flows to absolute bandwidth thresholds.

- **Bandwidth Piracy**—Bandwidth policing ensures that sessions consume no more bandwidth than what is signaled for.

Policing Methods

The Net-Net SBC polices real-time traffic by using Constant Bit Rate (CBR) media policing. CBR policing is used when a media flow requires a static amount of bandwidth to be available during its lifetime. CBR policing best supports real-time applications that have tightly constrained delay variation. For example, voice and video streaming are prime candidates for CBR policing.

Session Media Flow Policing

Session media encompasses RTP and RTCP flows. In order to select policing constraints for these flows, the Net-Net SBC watches for the codec specified in an SDP or H.245 message. When a match is made between the codec listed in an incoming session request and a configured media profile, the Net-Net SBC applies that media profile's bandwidth policing constraint to the media flow about to start.

If multiple codecs are listed in the SDP message, the Net-Net SBC will use the media profile with the most permissive media policing constraints for all of the flows associated with the session. If a codec in the H.245/SDP message is not found in any configured media profile, the Net-Net SBC uses the media profile with the most permissive media policing constraints configured. If no media profiles are configured, there will be no session media flow policing.

If a mid-call change occurs, bandwidth policing is renegotiated.

Static Flow Policing

Static flows can also be policed in the same way as media flows are policed. A static flow configuration redirects flows entering the Net-Net SBC on a front interface. The redirection is based on realm, source, destination, and protocol. When a flow matches the configured static flow criteria, besides being redirected toward a specified destination, its rate can also be controlled based on a static flow policing parameter. Static flow policing operates obliviously to the data contained within the flow.

Configuration Notes

Review the following information before configuring your Net-Net SBC to perform media policing.

Session Media Flow Policing

Session media flow policing applies to both RTP and RTCP flows. Setting either of the parameters listed below to 0 disables media policing, letting RTP or RTCP flows pass through the Net-Net SBC unrestricted.

- **RTP Policing**
 - Set in the media profile's average rate limit parameter to police RTP traffic with the CBR policing method.
 - Average rate limit parameter establishes the maximum speed for a flow in bytes per second.
- **RTCP Policing**
 - Set in the media manager configuration. The RTCP rate limit is set to police RTCP traffic with the CBR policing method.

- RTCP rate limit parameter establishes the maximum speed for an RTCP flow in bytes per second.

Static Flow Policing

Static flow policing is configured with one parameter found in the static flow configuration. To configure CBR, you have to set the average rate limit to a non-zero value. Setting the parameter listed below to 0 disables static flow policing, effectively letting the flow pass through the Net-Net SBC unrestricted.

In a CBR configuration, the average rate limit determines the maximum bandwidth available to the flow.

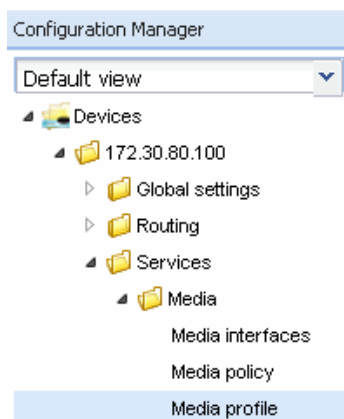
- Average rate limit establishes the maximum speed for a static flow in bytes per second.

Note: Static flow policing is not necessarily tied to any type of media traffic, it can affect flows of any traffic type.

Configuring Media Policing for RTP Flows

You can configure media policing for RTP flows.

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Select the Media profile you want to configure and click **Edit**.
4. Scroll to Policing.
5. **Maximum bandwidth allowed per flow (bytes per second)**—Enter the maximum rate up to 125Mbytes per second for RTCP packet flow.

Maximum bandwidth allowed per flow
(bytes per second), Media policing

- Minimum: 0 (media policing disabled)
- Maximum: 125000000
- Default: 0

Average rate limit values for common codecs:

- PCMU—80000 Bps
- G729—26000 Bps

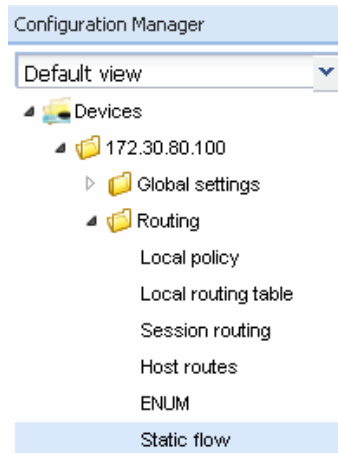
6. Click **Apply**.

Configuring Media Policing for Static Flows

You can configure media policing for static flows using the Net-Net Central.

To configure media policing for static flows:

1. Expand the Configuration Manager slider > Routing.



2. Click Static flow. The Static flows table appears in the content area.
3. Select a configuration from the table and click **Edit**.
4. **Maximum bandwidth allowed per flow (bytes/sec)**—Enter the maximum sustained rate limit in bytes per second.

Flow policing

Maximum bandwidth allowed per flow

- Minimum: 0 (media policing disabled)
- Maximum: 125000000

5. Click **Apply** to complete configuring this static flow.

RTP Payload Type Mapping

The Net-Net SBC maintains a default list of RTP payload types mapped to textual encoding names as defined in RFC 3551.

The following table defines the preconfigured payload type for standard encodings.

Payload Type	Encoding Name	Audio (A) / Video (V)	Clock Rate
0	PCMU	A	8000
4	G723	A	8000
8	PCMA	A	8000
9	G722	A	8000
15	G728	A	8000
18	G729	A	8000

If you configure any payload type to encoding name mappings, the default mappings will be ignored. You must then manually enter all payload type mappings you use in the **media-profile** configuration element.

ITU-T to IANA Codec Mapping

The Net-Net SBC maintains a list of ITU-T (H.245) codecs that map to IANA RTP codecs. An ITU codec is directly mapped to an IANA Encoding Name for media profile lookups. All codecs are normalized to IANA codec names before any matches are made. New ITU-T codecs can not be added to the media profiles list.

The following table defines the ITU-T to IANA codec mappings.

ITU-T	IANA
g711Ulaw64k	PCMU
g711Alaw64k	PCMA
g726	G726
G7231	G723
g728	G728
g729wAnnexB	G729
g729	G729 fmt:18 annexb=no
H261VideoCapability	H261
H263VideoCapability	H263
t38Fax	T38

TCP Synchronize Attack Prevention

This section explains how the Net-Net SBC protects itself from a Transmission Control Protocol (TCP) synchronize (SYN) packet flooding attack sourced from a remote hostile entity.

SIP and H.323 signaling can be configured on the Net-Net SBC to be TCP protocol-based. In this configuration, the Net-Net SBC can be a target of a TCP SYN attack. The Net-Net SBC is able to service new call requests throughout the duration of an attack.

About SYN

SYN is used by TCP when initiating a new connection to synchronize the sequence numbers on two connecting computers. The SYN is acknowledged by a SYN-ACK by the responding computer. After the SYN-ACK, the client finishes establishing the connection by responding with an ACK message. The connection between the client and the server is then open, and the service-specific data can be exchanged between the client and the server.

A SYN flood is a series of SYN packets from forged IP addresses. The IP addresses are chosen randomly and do not provide any hint of the attacker's location. The SYN flood keeps the server's SYN queue full. Normally this would force the server to drop connections. A server that uses SYN cookies, however, will continue operating normally. The biggest effect of the SYN flood is to disable large windows.

Server Vulnerability

Vulnerability to attack occurs when the server has sent a SYN-ACK back to client, but has not yet received the ACK message; which is considered a half-open connection. The server has a data structure describing all pending connections built

in its system memory. This data structure is of finite size, and it can be made to overflow by intentionally creating too many partially-open connections.

The attacking system sends SYN messages to the server that appear to be legitimate, but in fact reference a client that is unable to respond to the SYN-ACK messages. The final ACK message is never sent to the server.

The half-open connections data structure on the server fills and no new incoming connections are accepted until the table is emptied out. Typically there is a timeout associated with a pending connection (the half-open connections will eventually expire and the server will recover). But the attacking system can continue sending IP-spoofed packets requesting new connections faster than the server can expire the pending connections. The server has difficulty in accepting any new incoming network connections.

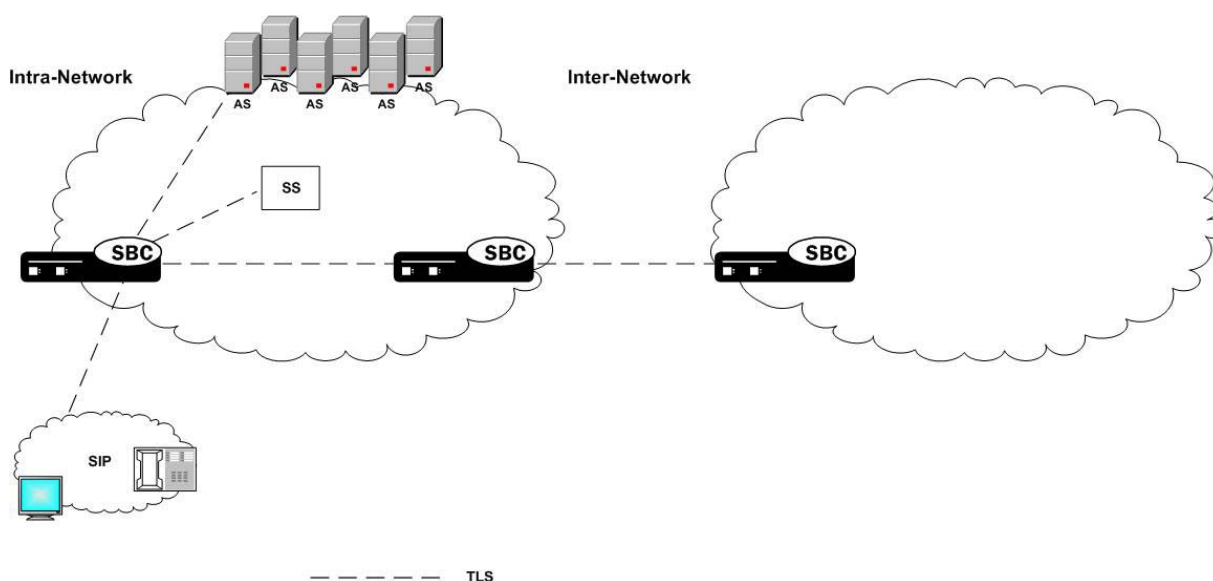
Configuring TCP SYN Attack Prevention

No configuration is necessary to enable TCP SYN attack prevention. Internal TCP protocol changes were made to provide protection.

Transport Layer Security

The Net-Net SBC provides support for Transport Layer Security (TLS) for SIP, which can be used to protect user and network privacy by providing authentication and guaranteeing the integrity for communications between the Net-Net SBC and the following:

- Another device in your network infrastructure (intra-network)
- Another Net-Net SBC when you are using a peering application (inter-network) for interior network signaling security
- An endpoint for authentication before allowing SIP messaging to take place



The Net-Net SBC and TLS

The Net-Net SBC's TLS functionality depends on the presence of a the Signaling Security Module (SSM) for hardware acceleration of encryption and decryption and random media generation. The SSM is a plug-on module that can be added to your Net-Net SBC chassis given the installation of the necessary bootloader and minimum hardware revision levels.

With the requisite hardware revision levels, the plug-on unit can be added to your Net-Net SBC in the field by qualified personnel. This provision makes upgrades fast, forgoing the need for you to return your Net-Net SBC to Acme Packet manufacturing for hardware upgrade. When your Net-Net SBC is upgraded with the SSM card that supports TLS, a new CLEI code will be added to your chassis; the code will also appear on the SSM card (also referred to as the plug-on unit) and visible if the system's chassis cover is opened. New Net-Net SBCs outfitted with the SSM card will have the code labels already affixed in all required locations.

TLS support will not behave in the manner described here if you do not have the SSM component installed on your Net-Net SBC, because it is the presence of this hardware that enables the TLS software support.

The accelerator card performs:

- RSA
- Diffie-Hellman
- DES
- 3DES
- 40/128 bit ARCFOUR
- AES256
- Random number generation

TLS Features

The Net-Net SBC supports the following TLS features:

- TLSv1/SSLv3
- RFC 3261 specific SIPs and TLS support in SIP
- Importing X509v3 certificates in PKCS-7/X509v3 PEM/Base64 format
- Generating a private key and a certificate request in PKCS-10 PEM/Base64 format
- Displaying imported certificates in text format
- Configuration verification, including verification that all dependencies are resolved
- Connection reuse draft (draft-ietf-sip-connect-reuse-03.txt)
- HA for TLS—When the active system in an HA node fails, the standby has the same TLS-related configuration, which is accomplished through configuration checkpointing as described in the *HA Nodes* chapter.
 - Existing active calls are not affected by a failover—Enduser experiences no interruption or disturbance in service. SIP signaling messages sent over the connection following failover do not impact the active call.
 - New calls, new TLS connections are be made
- Real-time reconfiguration (RTC)

The Net-Net SBC does not support certificate revocation listing handling.

Domestic and International Versions

There are two versions of the Net-Net OS that support TLS: a U.S. version and an international version. Two versions exist because of the laws governing the strength of algorithms that can be shipped domestically and internationally. If you require further information, consult with your Acme Packet sales representative directly.

Compliance

The Net-Net SBC supports:

- FIPS and non-FIPS mode
- EAL4

Supported Encryption

The Net-Net SBC provides support for TLSv1 and SSLv3 encryption.

TLSv1 Ciphers

The Net-Net SBC supports the TLS v1 cipher suites listed in this section.

For encryption, the Net-Net SBC supports: AES-128, AES-256, 3DES, DES and ARC4 (40 and 128 bit) algorithms. It also supports:

- TLS_RSA_WITH_NULL_MD5
- TLS_RSA_WITH_NULL_SHA
- TLS_RSA_WITH_RC4_128_MD5
- TLS_RSA_WITH_RC4_128_SHA
- TLS_RSA_WITH_DES_CBC_SHA
- TLS_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_DHE_RSA_WITH_DES_CBC_SHA
- TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA
- TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
- TLSv1
- SSLv3
- ALL [default]
- NONE

Mapping SSL3 to TLSv1 Ciphers

The following table shows the mapping of SSL3 ciphers to TLSv1 ciphers:

SSL3	TLSv1
SSL_RSA_WITH_NULL_MD5	TLS_RSA_WITH_NULL_MD5
SSL_RSA_WITH_NULL_SHA	TLS_RSA_WITH_NULL_SHA
SSL_RSA_WITH_RC4_128_MD5	TLS_RSA_WITH_RC4_128_MD5
SSL_RSA_WITH_RC4_128_SHA	TLS_RSA_WITH_RC4_128_SHA
SSL_RSA_WITH_DES_CBC_SHA	TLS_RSA_WITH_DES_CBC_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA	TLS_RSA_WITH_3DES_EDE_CBC_SHA

SSL3	TLSv1
SSL_DHE_RSA_WITH_DES_CBC_SHA	TLS_DHE_RSA_WITH_DES_CBC_SHA
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA	TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA

Note: The Net-Net SBC supports **TLS_RSA_WITH_NULL_MD5** and **TLS_RSA_WITH_NULL_SHA** although neither does any encryption. These ciphers might be used for debugging purposes, and should not be deployed.

Signaling Support

The Net-Net SBC's TLS functionality supports SIP and SIPS. In addition, the Net-Net SBC can accommodate a mixture of TLS and non-TLS sessions within a realm as because a request for TLS is controlled by the endpoint (TLS UA).

DoS Protection

The Net-Net SBC provides the following forms of DoS protection from:

- Too many simultaneous TLS connections being requested by a single IP address.

The Net-Net SBC limits the number of TLS connections from a single IP address; you can set a maximum simultaneous number of TCP/TLS connections a SIP interface will allow from a single IP address.

- Too many simultaneous TLS connections being requested by limiting the maximum number of connections for a SIP interface.

In other words, the maximum simultaneous TCP/TLS connections a SIP interface will allow in aggregate from all IP addresses served by that signaling interface.

- Endpoints establishing TCP/TLS connections that never send any messages (application layer messages; once the TLS handshake completes).

This protection is triggered by inactivity, measured by lack of any message from a peer. The value specified for this timer is in seconds.

- Endpoints requesting an initial registration that never send messages thereafter.

Note: It is expected that whenever an endpoint establishes a TCP/TLS connection, it will keep the connection active by sending additional messages or by using the NAT interval configuration. Whenever a connection is torn down because of inactivity, a log at the level "ERROR" is generated.

- Malformed packets by counting and limiting the maximum number of malformed packets.

Whenever the Net-Net SBC receives an invalid TLS message, it increments the internal invalid signalling threshold counter. When that counter reaches the configured value, the Net-Net SBC denies the endpoints for the configured deny period. This also requires configuration of tolerance window in media manager.

Endpoint Authentication

The Net-Net SBC does not operate as a CA. Instead, the Net-Net SBC's TLS implementation assumes that you are using one of the standard CAs for generating certificates:

- Verisign
- Entrust
- Thawte
- free Linux-based CA (for example, openssl)

The Net-Net SBC can generate a certificate request in PKCS10 format and to export it. It can also import CA certificates and a Net-Net SBC certificate in the PKCS7/X509 PEM format.

The Net-Net generates the key pair for the certificate request internally. The private key is stored as a part of the configuration in 3DES encrypted form (with an internal generated password) and the public key is returned to the user along with other information as a part of PKCS10 certificate request.

The Net-Net SBC supports the option of importing CA certificates and marking them as trusted. However, the Net-Net SBC only authenticates client certificates that are issued by the CAs belonging to its trusted list. If you install only a specific vendor's CA certificate on the Net-Net SBC, it authenticates that vendor's endpoints. Whether the certificate is an individual device certificate or a site-to-site certificate does not matter because the Net-Net SBC authenticates the signature/public key of the certificate.

Viewing TLS Support Information

Currently you need to configure TLS support using the ACLI, you can view TLS support information using the Net-Net Central. See the *Net-Net 4.1 Configuration Guide* for ACLI details.

ACLI Process Overview

In summary, you need to take the following steps to enable your Net-Net SBC for TLS.

1. Make sure that your Net-Net SBC has the appropriate hardware installed and that you have obtained an enabled the licenses related to TLS support.
2. Configure certificates.
3. Configure the specific parameters related to TLS.

Configuring Certificates using the ACLI

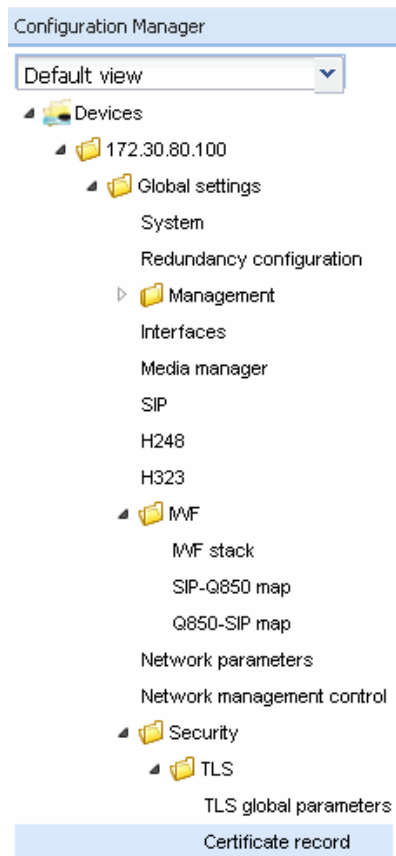
Configuring certificates is a three-step process:

1. Create a certificate record configuration on the Net-Net SBC
2. Generate a certificate request by the Net-Net SBC and save the configuration
3. Import the certificate record into the Net-Net SBC and save the configuration

Viewing TLS Certificate Records

To view a certificate record:

1. Expand the Configuration Manager slider > Global settings > Security > TLS.



2. Click Certificate record.
3. In the Certificate record table, click a row to select the certificate record.

4. Select a record and click **View**. The certificate record appears in the content area. You cannot make any edits to this record.

*Name	name0
Country	US
State	MA
Locality	Burlington
Organization	Engineering
Unit	
Common name	common
Alternate name	
Key size	1024
Trusted	enabled

5. Click **Back** to close the window.

Generating a Certificate Request

Using the ACLI **generate-certificate-request** command allows you to generate a private key and a certificate request in PKCS10 PEM format. You take this step once you have configured a certificate record.

The Net-Net SBC stores the private key that is generated in the certificate record configuration in 3DES encrypted form with an internally generated password. The PKCS10 request is displayed on the screen in PEM (Base64) form.

You use this command for certificate record configurations that hold end-entity certificates. If you have configured the certificate record to hold a CA certificate, then you do not need to generate a certificate request because the CA publishes its certificate in the public domain. You import a CA certificate by using the ACLI **import-certificate** command.

This command sends information to the CA to generate the certificate, but you cannot have Internet connectivity from the Net-Net SBC to the Internet. You can reach the Internet through a browser such as Internet Explorer if it is available, or you can save the certificate request to a disk and then submit it to the CA.

To run the applicable command, you must use the value you entered in the name parameter of the certificate record configuration. You run the command from main Superuser mode command line:

```
ACMEPACKET# generate-certificate-request acmepacket
Generating Certificate Signing Request. This can take several
minutes...

-----BEGIN CERTIFICATE REQUEST-----
MIIDHzCCAoigAwIBAgIIAhMCUACEAHewDQYJKoZIhvcNAQEFBQAwDELMakGA1UE
BhMCVVMxEzARBgNVBAgTCKNhbgLmb3JuaWExETAPBgNVBACTCFNhb3NlMkQ4w
DAYDVQQKEwVzaXBpdDEpMCCGA1UECzMgU2lwaXQgVGVzdCBBDXJ0awZpY2F0ZSBB
dXRob3JpdHkwHhcNMjUwNDEzMjEzNzQzWcNMDgwNDEyMjEzNzQzWjBUMQswCQYD
VQQGEwJVUzELMakGA1UECBMCTUEwEzARBgNVBAgTCKJ1cmxpbnR0b24xZDASBgNV
BAoTC0Vuz2luZWVyaW5nMQowCwYDVQQDEwRhY21lMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCXjIeOyFKAUB3rKkKk/+59LT+rLGUw7Lgc1V6+hFTSr0co+ZsQ
bHFUWAA15qXUUBTLJG13QN5VfG96f7gGAbWayf0S9Uymold3JPCUD0Ggb2E7m8iu
vtq7gwjSeKNXAw/y7ywy/c04FmUD2U0pZX0CNIR3Mns50AxQmq0bNYDhawIDAQAB
o4HdMIHaMBEGA1UdEQKMAiCBnBrdw1hcjAJBgNVHRMEAjAAMB0GA1UdDgQWBBTG
tpodxa6Kmmn04L3Kg62t8BZJHTCBmgYDVR0jBIGSMIGPgBRrRhcU6pR2JYBubhNU
2qHjVBShtqF0pHIwcDELMakGA1UEBhMCVVMxEzARBgNVBAgTCKNhbgLmb3JuaWEx
ETAPBgNVBACTCFNhb3NlMkQ4wDAYDVQQKEwVzaXBpdDEpMCCGA1UECzMgU2lwaXQg
VGVzdCBBDXJ0awZpY2F0ZSBBdXRob3JpdHkMAQAwDQYJKoZIhvcNAQEFBQAD
gYEAbEs8nUCi+cA2hC/LM49Sitvh8Qmpl81KONApsoC4Em24L+DZwz3uInoWbjJ
QhefcUfteNYkbuMH7LAK0hndPvW+St4rQGVK6LJhZj7/yeLXmYWIPIUY3Ux40Gvrd
2UgV/B2S0qH9Nf+FQ+mNZ0LL7EuF4IxSz9/69LuYlXqKsG4=
-----END CERTIFICATE REQUEST-----;

WARNING: Configuration changed, run "save-config" command.
ACMEPACKET# save-config
Save-config received, processing.
waiting 1200 for request to finish
Request to 'SAVE-CONFIG' has Finished,
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot-activate'
ACMEPACKET# activate-config
Activate-Config received, processing.
```

```

waiting 12000 for request to finish
Add LI flows
LiSysClientMgr::handleNotifyReq
H323 Active Stack Cnt: 0
Request to 'ACTIVATE-CONFIG' has finished
Activate Complete
ACMEPACKET#

```

Importing a Certificate Using the ACLI

For an end-entity certificate, once a certificate is generated using the ACLI generate-certificate-request command, that request should be submitted to a CA for generation of a certificate in PKCS7 or X509v3 format. When the certificate has been generated, it can be imported into the Net-Net SBC using the **import-certificate** command.

The syntax is:

```
ACMEPACKET # import-certificate [try-all|pkcs7|x509] [certificate-record file-name]
```

To import a certificate:

1. When you use the **import-certificate** command, you can specify whether you want to use PKCS7 or X509v3 format, or try all. In the command line, you enter the command, the format specification, and the name of the certificate record.

```
ACMEPACKET# import-certificate try-all acme
```

The following will appear:

Please enter the certificate in the PEM format.

Terminate the certificate with ";" to exit.....

```

-----BEGIN CERTIFICATE-----
MIIDHzCCAoigAwIBAgIIAhMCUACEAHewDQYJKoZIhvcNAQEFBQAwDELMAKGA1UE
BhMCVVMxEzARBgNVBAGTCkNhbgGmb3JuawExETAPBgNVBACTCFNhbiBkb3NlMQ4w
DAYDVQQKEwVzaXBpdDEpMCCGA1UECXMgU2lwaXQgVGVzdCBDZXJ0awZpY2F0ZSBB
dXRob3JpdHkwHhcNMDUwNDEzMjEzNzQzWhcNMDgwNDEyMjEzNzQzWjBUMQswCQYD
VQQGEwJVUzELMAKGA1UECBMCTUExEzARBgNVBACTCkU1cmxpbmd0b24xFDASBgNV
BAoTC0Vuz2luZWVyaW5nMQ0wCwYDVQQDEwRhY21lMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCXjIeOyFKAUB3rKkKK/+59LT+rLGUw7Lgc1V6+hfTSr0co+ZsQ
bHFUWAA15qXUUBTLJG13QN5VF696f7gGAbWayf0S9Uymo1d3JPCUD0Ggb2E7m8iu
vtq7gwjSeKNXAw/y7ywy/c04FmUD2U0pZX0CNIR3Mns50AxQmq0bNYDhawIDAQAB
o4HdMIHaMBEGA1UdEQQKMAiCBnBrdw1hcjAJBgNVHRMEAjAAMB0GA1UdDgQWBBTG
tpodxa6Kmmn04L3Kg62t8BZJHTCBmgYDVR0jBIGSMIGPgBRrRhcU6pR2JYBUBhNU
2qHjVBShtqF0pHIwcDELMAKGA1UEBhMCVVMxEzARBgNVBAGTCkNhbgGmb3JuawEx
ETAPBgNVBACTCFNhbiBkb3NlMQ4wDAYDVQQKEwVzaXBpdDEpMCCGA1UECXMgU2lwa
XQgVGVzdCBDZXJ0awZpY2F0ZSBBdXRob3JpdHmCAQAwDQYJKoZIhvcNAQEFBQAD
gYEAbEs8nUCi+cA2hC/LM49Sitvh8QmpL81KONApsoC4Em24L+DZwz3uInowbjbJJ
QhefcUfteNYkbuMH7LAK0hndPvW+St4rQGVK6LJhZj7/yeLXmYWIPIUY3Ux40GVrd
2UgV/B2S0qH9Nf+FQ+mNZ0LL7EuF4IxSz9/69LuYlXqKsG4=
-----END CERTIFICATE-----;

```

Certificate imported successfully....

WARNING: Configuration changed, run "save-config" command.

2. Save your configuration.

```
ACMEPACKET# save-config
Save-Config received, processing.
waiting 1200 for request to finish
Request to 'SAVE-CONFIG' has Finished,
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot activate'.
```

3. Synchronize and activate your configurations.

```
ACMEPACKET# activate-config
Activate-Config received, processing.
waiting 120000 for request to finish
Add LI Flows
LiSysClientMgr::handleNotifyReq
H323 Active Stack Cnt: 0
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
ACMEPACKET#
```

Importing a Certificate Using FTP

You can also put the certificate file in the directory /ramdrv and then executing the **import-certificate** command or by pasting the certificate in the PEM/Base64 format into the ACLI. If you paste the certificate, you might have to copy and paste it a portion at a time rather than pasting in the whole thing at once.

To import the certificate using FTP:

1. FTP the certificate file on to the Net-Net SBC (directory /ramdrv), let us say the name of the certificate file is **cert.pem**.
2. Once the certificate is successfully transferred to the Net-Net SBC, run the **import-certificate** command.

The syntax is:

```
ACMEPACKET# import-certificate [try-all|pkcs7|x509] [certificate-record file-name]
```

Using the command will look like this when you have used FTP.

```
ACMEPACKET# import-certificate try-all acme cert.pem
Certificate imported successfully...
WARNING: Configuration changed, run "save-config" command.
```

3. Save your configuration.

```
ACMEPACKET# save-config
Save-Config received, processing.
waiting 1200 for request to finish
Request to 'SAVE-CONFIG' has Finished,
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot activate'.
```

4. Synchronize and activate your configurations.

```
ACMEPACKET# activate-config
Activate-Config received, processing.
waiting 120000 for request to finish
Add LI Flows
LiSysClientMgr::handleNotifyReq
H323 Active Stack Cnt: 0
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
ACMEPACKET#
```

Configuring a TLS Profile

The TLS profile contains the information required to run SIP over TLS. You can configure a TLS profile using the ACLI. You can only view TLS profile information with the Net-Net Central.

ACLI Instructions and Examples

To configure a TLS profile:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **security** and press <Enter> to access the **session-router** path.


```
ACMEPACKET(configure)# security
```
3. Type **tls-profile** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(security)# tls-profile
ACMEPACKET(tls-profile)#
```
4. **name**—Enter the name of the TLS profile. This parameter is required; you cannot leave it empty.
5. **end-entity-cert**—Enter the name of the entity certification record.
6. **trusted-records**—Enter the names of the trusted CA certificate records.
7. **cipher-list**—Either use the default **ALL**, or enter a list of ciphers you want to support. Refer to the Supported Encryption section of this chapter for more details.
8. **verify-depth**—Specify the maximum depth of the certificate chain that will be verified. Use the default of **5**, or enter the numeric value your configuration requires.
9. **mutual-authenticate**—Define whether or not you want the Net-Net SBC to mutually authenticate the client. Leave this parameters set to its default of **disabled**, or type in **enabled** to turn on mutual authentication.
10. Save your work.
11. Exit out to the configuration terminal menu to apply the TLS profile.

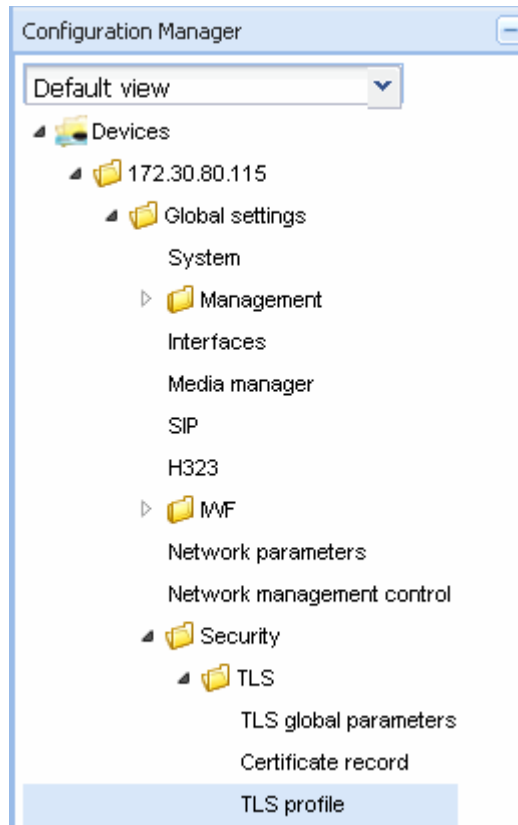

```
ACMEPACKET(tls-profile)# exit
ACMEPACKET(security)# exit
ACMEPACKET(configure)#
```

Viewing TLS Profiles using Net-Net Central

Currently you need to configure TLS support using the ACLI, you can view TLS support information using the Net-Net Central.

To view a TLS profile:

1. Expand the Configuration Manager slider > Global settings > Management > Security > TLS.



2. Click TLS profile.
3. In the Tls profile table, click a row to select the profile.

- Click **View**. The certificate record appears in the content area. You cannot make any edits to this record.

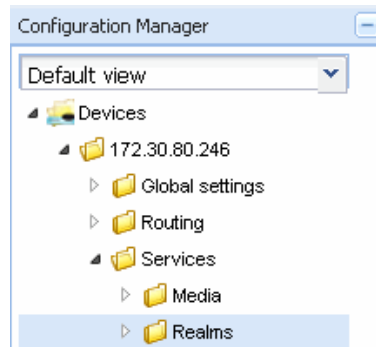
*Name	tls-prfl1
Verify depth	10
Mutual authenticate	disabled
End entity Certificate	name0
TLS version	tlsv1
Cipher	ALL Add Delete
Trusted	mytest Add Delete

- Click **Back** to close the window.

Applying a TLS Profile

To apply the TLS profile, you need to specify it for the SIP interface with which it will be used. You must take this step from within the SIP interface configuration.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. Scroll to the SIP signaling interface table, select a SIP interface and click **Edit**.
7. **TLS profile**—Choose the profile name from the TLS profile combination box.

TLS profile

8. Click **Apply** to continue.

Reusing a TLS Connection

The Net-Net SBC supports TLS connection reuse if and when an **alias** is included in the Via header by the originator of the TLS connection. When this is the case, the Net-Net SBC reuses the same connection for any outgoing request from the Net-Net SBC.

Keeping Pinholes Open at the Endpoint

The Net-Net SBC provides configurable TCP NAT interval on a per-realm basis. You need to configure a NAT interval for the applicable realm to support either all conforming or all non-conforming endpoints.

- Conforming endpoints use the draft-jennings sipping-outbound-01. It describes how to keep the endpoint keeps the connection alive.

Note: Currently the endpoint uses REGISTER.

- Non-conforming endpoints have short NAT interval, where the HNT application with the TCP connection for TLS operates as it does for regular TCP. We give the UA a shorter expires time so that it refreshes frequently, implicitly forcing the UA to keep the TVP socket open and reuse it for further requests (in-dialog or out-of-dialog). Regular requests using TLS sent from the Net-Net SBC to the UA reuse the same TCP connection so that further TLS certificate exchanges are not required.

Viewing Certificates

You can view either a brief version or detailed information about the certificates.

Brief Version

Obtaining the brief version uses this syntax, and will appear like the following example:

```
ACMEPACKET# show certificates brief acmepacket

certificate-record:acmepacket

Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      02:13:02:50:00:84:00:71
    Issuer:
      C=US
      ST=California
      L=San Jose
      O=sipit
      OU=Sipit Test Certificate Authority
    Subject:
      C=US
      ST=MA
      L=Burlington
      O=Engineering
      CN=acme
ACMEPACKET#
```

Detailed Version

Obtaining the detailed version uses this syntax, and will appear like the following example:

```
ACMEPACKET#show certificates detail acmepacket

certificate-record:acmepacket

Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      02:13:02:50:00:84:00:71
    Signature Algorithm: sha1WithRSAEncryption
    Issuer:
      C=US
      ST=California
      L=San Jose
      O=sipit
      OU=Sipit Test Certificate Authority
    Validity
      Not Before: Apr 13 21:37:43 2005 GMT
      Not After : Apr 12 21:37:43 2008 GMT
    Subject:
      C=US
      ST=MA
```

```

L=Burlington
O=Engineering
CN=acme
X509v3 extensions:
  X509v3 Subject Alternative Name:
    DNS:pkumar
  X509v3 Basic Constraints:
    CA:FALSE

```

```
ACMEPACKET#
```

Denial of Service for TLS

This section explains the DoS for TLS feature. With this feature, the Net-Net SBC can provide protection from TCP/TLS message flood by limiting the number of connections from an end point and by limiting the number of simultaneous TCP/TLS connections to a SIP interface.

The Net-Net SBC protects against a flood of invalid TLS messages and against end points establishing TCP/TLS connections or doing an initial registration without then sending any messages. The Net-Net SBC protects against:

- Too many simultaneous TLS connections being requested by a single IP address by limiting the number of TLS connections from a single IP address. There is a maximum simultaneous number of TCP/TLS connections a SIP interface will allow from a single IP address.
- Too many simultaneous TLS connections being requested by limiting the maximum number of connections for a SIP interface. There is a maximum number of simultaneous TCP/TLS connections a SIP interface will allow in aggregate from all IP addresses served by that signaling interface.
- End points establishing TCP/TLS connections without then sending any messages (application layer messages post TLS handshake complete). Triggered by inactivity as measured by lack of any message from this peer.
- End points doing an initial registration without then sending any messages.

This timer could be used by the administrator to detect errors with the SIP configuration. It is expected that whenever an end point establishes a TCP/TLS connection, the end point will keep the connection active by sending messages with REGISTER or by using the NAT interval configuration. Whenever a connection is torn down because of inactivity, a log at the level ERROR is generated.)

- Malformed packets by counting and limiting the maximum number of malformed packets. Whenever an invalid TLS message is received, the internal counter corresponding to invalid-signal-threshold is incremented. When the invalid signal threshold reaches the configured value, the end point will be denied for the configured deny period. (Also requires configuration of the tolerance window in media manager.)

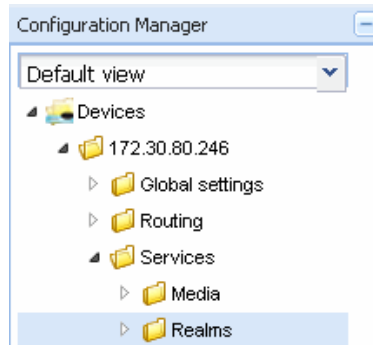
Configuring DoS for TLS

You configure the SIP interface and the realm to support DoS for TLS.

Configuring the SIP Interface

To configure the SIP interface:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab. The Realm services table appears.
5. Select a SIP interface service in the table and click **Edit**.
6. **Maximum number of connections, Per source IP address**—Enter the maximum number of connections allowed from an end point. The range of values is 0 to SBC0. The default value of 0 disables this parameter.

Note: To make this parameter effective, you need to set the realm's access-control-trust-level to low or medium.

7. **Maximum number of connections, Per SIP interface**—Enter the maximum number of simultaneous TCP/TLS connections for this SIP interface. The range of values is 0 to SBC0. The default value of 0 disables this parameter.
8. **Inactivity timeout per connection (sec), Inactivity timeout**—Enter the time in seconds you want a connection from an endpoint discontinued. This provides protection from end points doing an initial registration without sending any messages. The default value is zero (0), which disables the timer.

9. **Untrusted timeout connection (sec), Inactivity timeout**—Enter the time in seconds you want a connection from an untrusted endpoint discontinued. This provides protection from end points doing an initial registration without sending any messages. The default value is zero (0), which disables the timer.

TCP/TLS Denial service protection

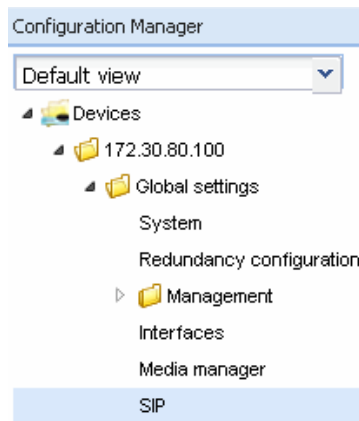
Maximum number of connections, Per source IP address	0
Maximum number of connections, Per SIP interface	0
Inactivity timeout per connection(sec), Inactivity timeout	0
Untrusted connection timeout (sec), Connection timeout	0

10. Click **Apply**.

Configuring SIP Configuration

To configure the SIP configuration in the Net-Net Central:

1. Expand the Configuration Manager slider > Global settings.



2. Click **SIP**.
3. Click **Add** to add the SIP configuration in the content area.
4. **Inactive TCP/TLS-based sessions timeout(sec)**—Enter the time in seconds after which if the peer does not send SIP messages after it initiates a TCP connection, the connection is torn down. This protects against endpoints establishing TCP/TLS connections and then not sending messages. The range of values is 0 to 999999999. The default value is 0, which disables the timer.

Inactive TCP/TLS-based session timeout (sec)	33

Because the Net-Net SBC first establishes a TCP connection, then the TLS connection it waits twice the value entered here after the initiation of a TLS connection before tearing down the connection.

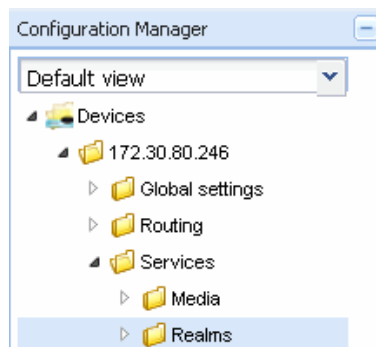
After an endpoint establishes a TCP/TLS connection, it is supposed to keep the connection active by sending messages or by using the NAT interval

configuration. Whenever a connection is torn down because of inactivity, a log at the level “ERROR” is generated.

Configuring the Realm

To configure the realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm in the table and click **Edit**.
4. Click the Access control tab. The Access control table appears in the content area.
5. Select an Access control configuration and click Edit.
6. **Deny period(s)**—Set the time period in seconds after which the entry for this host is removed from the deny list.
 - minimum: zero (0) seconds
 - maximum: 4294967295 seconds

The default value is 30 seconds.

Deny period(s)

7. **Trust level**—Set the trust level for the host within the realm.
 - none: host is always untrusted. It is never promoted to the trusted list or demoted to the deny list.
 - low: host can be promoted to the trusted list or demoted to the deny list.
 - medium: host can be promoted to the trusted list but is only demoted to untrusted. It is never added to the deny list.
 - high: host is always trusted.

Trust level

8. Click **Apply**.

TLS Key Usage and Other Enhancements

This section describes a body of enhancements to the Net-Net SBC's TLS support. You can:

- Control the role of the certificate based on the usage model
- Set the TLS version, including the deprecation of the **Cipher list** parameter values TLSv1 and SSLv3

A parameter called **options** has been reserved for future enhancements both in the TLS profile and the certificate record configurations.

Key Usage Control

You can configure the role of a certificate by setting key usage extensions and extended key usage extensions. Both of these are configured in the certificate record configuration.

Key Usage List

This section defines the values you can use (as a list) in the **Key usage extensions** parameter. You can configure the parameter with more than one of the possible values.

Value	Description
digitalSignature (default with keyEncipherment)	Used when the subject public key is used with a digital signature mechanism to support security services other than non-repudiation, certificate signing, or revocation information signing. Digital signature mechanisms are often used for entity authentication and data origin authentication with integrity.
nonRepudiation	Used with the subject public key is used to verify digital signatures that provide a non-repudiation service protecting against the signing entity falsely denying some action, excluding certificate or CRL signing.
keyEncipherment (default with digitalSignature)	Used with the subject public key is used for key transport. (For example, when an RSA key is to be used for key management.)
dataEncipherment	Used with the subject public key is used for enciphering user data other than cryptographic keys.
keyAgreement	Used with the subject public key is used key agreement. (For example, when a Diffie-Hellman key is to be used for a management key.)
encipherOnly	The keyAgreement type must also be set. Used with the subject public key is used only for enciphering data while performing key agreement.
decipherOnly	The keyAgreement type must also be set. Used with the subject public key is used only for deciphering data while performing key agreement.

Extended Key Usage List

This section defines the values you can use in the **Extended key usage extensions** parameter.

Value	Description
serverAuth (default)	Used while the certificate is used for TLS server authentication. In Net-Net SBC access-side deployments, the Net-Net SBC typically acts as a TLS server accepting TLS connections. You might use this setting while generating the end-entity-cert.
clientAuth	Used while the certificate is used for TLS client authentication. In Net-Net SBC core-side deployments, the Net-Net SBC typically acts as a TLS client initiating TLS connections. You might use this setting while generating the end-entity-cert.

Configuring Key Usage Control

You configure key usage control in the ACLI.

To configure the key-usage-list and the extended-key-usage-list parameters:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)#
```
2. Type **security** and press <Enter>.


```
ACMEPACKET(configure)# security
ACMEPACKET(security)#
```
3. Type **certificate-record** and press <Enter>. If you are adding this feature to a pre-existing certificate record configuration, you must select that configuration to edit it.


```
ACMEPACKET(security)# certificate-record
ACMEPACKET(certificate-record)#
```
4. **key-usage-list**—Enter the usage extensions you want to use with this certificate record. This parameter can be configured with multiple values, and it defaults to the combination of **digitalSignature** and **keyEncipherment**. For a list of possible values and their descriptions, refer to the Key Usage List section.
5. **extended-key-usage-list**—Enter the extended key usage extensions you want to use with this certificate record. The default is **serverAuth**. For a list of possible values and their descriptions, refer to the Extended Key Usage section.
6. Save and activate your configuration.

TLS Version

This section shows you how to configure the TLS version. Since you can control the TLS version with a parameter, the TLSv1 and SSLv3 values have been deprecated from the **cipher-list** parameter. The cipher list now support these values:

- TLS_RSA_WITH_NULL_MD5
- TLS_RSA_WITH_NULL_SHA
- TLS_RSA_WITH_RC4_128_MD5
- TLS_RSA_WITH_RC4_128_SHA
- TLS_RSA_WITH_DES_CBC_SHA
- TLS_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_DHE_RSA_WITH_DES_CBC_SHA

- TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA
- TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
- ALL (default)
- NONE

Configuring the TLS Version

You configure the TLS version in the ACLI.

To set the TLS version:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)#
```
2. Type **security** and press <Enter>.


```
ACMEPACKET(configure)# security
ACMEPACKET(security)#
```
3. Type **tls-profile** and press <Enter>. If you are adding this feature to a pre-existing TLS profile configuration, you must select that configuration to edit it.


```
ACMEPACKET(security)# tls-profile
ACMEPACKET(tls-profile)#
```
4. **key-usage-list**—Enter the TLS version you want to use with this TLS profile. Valid values are **TLSv1**, **SSLv3**, and **compatibility** (default).
5. Save and activate your configuration.

TLS Session Caching

TLS session caching lets the Net-Net SBC cache key information for TLS connections and set a timer for the length of time that the information will be cached.

Without this feature enabled, when the Net-Net SBC and a TLS client perform the handshake portion of the authentication sequence, they exchange a shared secret and encryption keys are generated. One of the results of the successful handshake is the creation of a unique session identifier. In the event that an established TLS connection is torn down but the client wants to reinstate it, this entire process must be repeated. Because the process is resource-intensive, the TLS session caching feature can be enabled to avoid repeating the handshake process for previously authenticated clients—thereby preserving valuable Net-Net SBC resources.

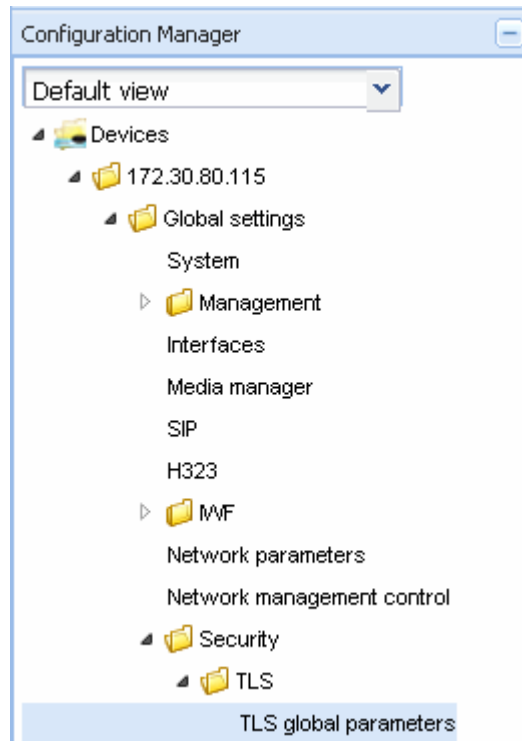
If TLS session caching is enabled on the Net-Net SBC, a formerly authenticated client can request re-connection using the unique session identifier from the previous session. The Net-Net SBC checks its cache, finds the session identifier, and reinstates the client. This reduces the handshake process to three messages. However, should the client offer an invalid session identifier (one that the Net-Net SBC has never seen, or one that has been deleted from its cache) the re-connection will fail, and will need to be negotiated from scratch.

Enabling TLS Session Caching

TLS session caching is global for all TLS functions on your Net-Net SBC.

To enable global TLS session caching:

1. Expand the Configuration Manager slider > Global settings > Management > Security > TLS.



2. Click TLS global parameters.
3. **Session caching**—Select enabled from the combination box for TLS session caching. The default is disabled.

Session caching

OCSP Support for SIP-TLS

Use these procedures to configure access to an Online Certificate Status Protocol (OCSP) responder. The OCSP is defined in RFC 2560, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*.

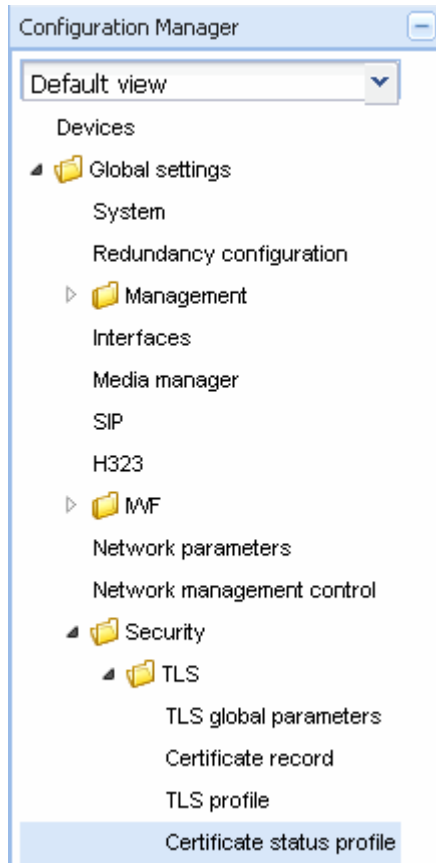
You must configure OCSP in the ACLI.

Viewing Certificate Status Profile Data

These parameters are configured in the ACLI. You can only view these parameters in Net-Net Central.

To view certificate status profile data:

1. Expand the Configuration Manager slider > Global settings > Security > TLS.



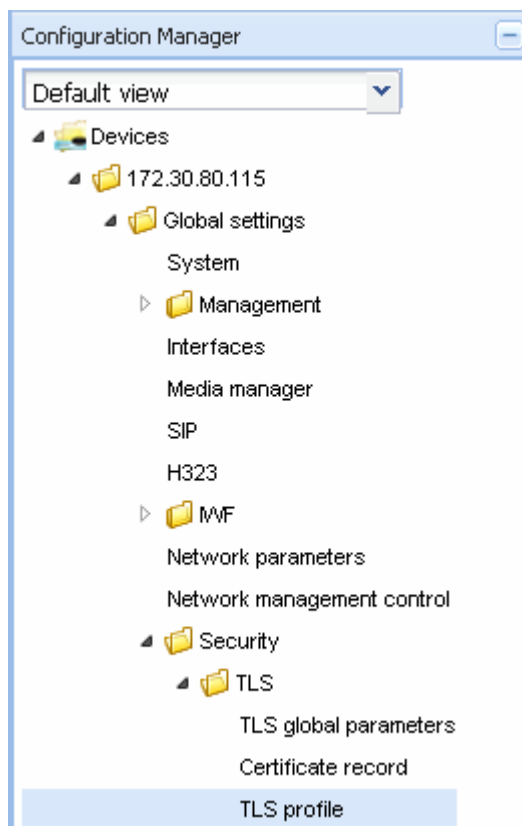
2. Click Certificate status profile. The Certificate status profile table appears in the content area.
3. Select a certificate status profile from the table and click **View**. You cannot make any edits to this record. Scroll to Certificate status profile.
4. Click **Cancel** to close the window.

Viewing TLS Profile Data

These parameters are configured in the ACLI. You can only view these parameters in Net-Net Central.

To view TLS certificate status check and certificate status profile:

1. Expand the Configuration Manager slider > Global settings > Management > Security > TLS.



2. Click TLS profile.
3. Select a TLS profile from the table and click **View**. You cannot make any edits to this record.
4. Click **Cancel** to close the window.

Password Strength

Password Policy

You can configure the minimum acceptable length for a password—between 8 and 64 characters—when password secure mode is enabled. For example, if you set this value to 15, then your password must be a minimum of 15 characters in length. The maximum password length is 64 characters (range is 8-64).

To increase password strength, new passwords must include at least one character from three out of four of the following character sets:

- Upper case letters
- Lower case letters
- Numbers
- Punctuation marks

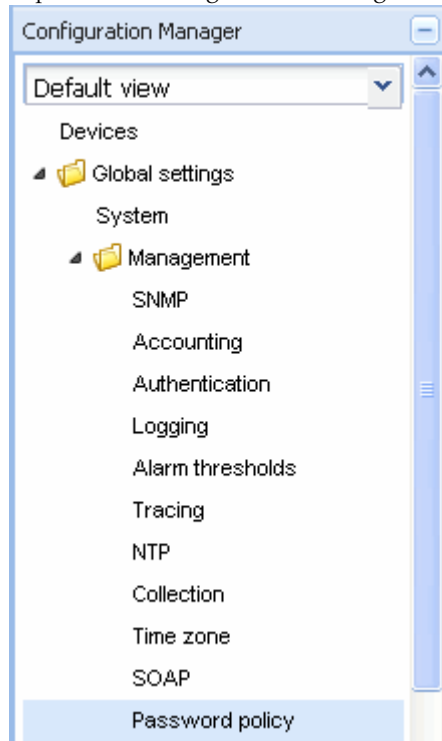
However, passwords cannot contain any of the following strings in any variation of case: `default`, `password`, `acme`, `user`, `admin`, `packet`.

Note: Any change you make to the password length requirement does not go into effect until you configure a new password. Pre-existing passwords can continue to be used until you change them.

Configuring Minimum Secured Password Length

To configure the minimum password length:

1. Expand the Configuration Manager slider.



2. Expand the Global settings folder, followed by the Management folder.
3. Click Password policy.
4. **Minimum password length in secure mode**—Enter a value between 8 and 64 characters that defines the minimum password length to use when in password secure mode. This parameter defaults to 8.

Password policy

Minimum password length in secure mode

5. Click **Apply**.

Setting a New Password

After setting a new minimum password length, you can change your password for either the User mode or the Superuser mode. The set password command is entered as follows:

```
set password [login | enable | reset]
```

Where

- login—Sets the User mode password
 - enable—Sets the Superuser mode password
 - reset—resets all passwords
1. Type **set password login** in User mode.
ACMEPACKET# set password login
 2. Enter your new password at the Enter new password prompt. Your typing will not be echoed to the screen.
Enter new password:
 3. If your password doesn't conform to the requirements, the ACLI prints a summary of your password's characteristics so you can see how to enter a valid password. For example:
ACMEPACKET# set password login
Enter new password:
% Summary:
% password length: 9
% alpha character count: 9
% numeric character count: 0
% upper-case character count: 0
% lower-case character count: 9
% punctuation character count: 0
% invalid character count: 0
%
% Only alphabetic (upper or lower case), numeric and punctuation
% character are allowed in the password.
% Password must be 10 - 64 characters,
% and have 3 of the 4 following character classes :
% - lower case alpha
% - upper case alpha
% - numerals
% - punctuation
Enter new password:
 4. Enter a new valid password, and then confirm the password.

Resetting Passwords

You can use the `set password [login | enable] reset` command to restore the associated password to the default.

System Access Control Lists

You can configure a system access control list (ACL) for your Net-Net SBC that determines what traffic the Net-Net SBC allows over its management interface (wancom 0). By specifying who has access to the Net-Net SBC via the management interface, you can provide DoS protection for this interface.

Using a list of IP addresses and subnets that are allowable as packet sources, you can configure what traffic the Net-Net SBC accepts and what it denies. All IP packets arriving on the management interface are subject; if it does not match your configuration for system ACL, then the Net-Net SBC drops it.

Note, however, that all IP addresses configured in the SNMP community table are automatically permitted.

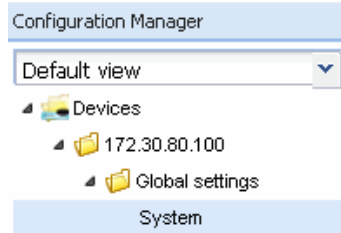
Configuring System Access Control Lists

For each ACL entry, you must define an IP destination address and mask. You can specify either the individual host or a unique subnet. If you do not configure this list, there will be no ACL/DoS protection for the Net-Net SBC's management interface.

Adding an ACL for the Management Interface

To add an ACL for the management interface:

1. Expand the Configuration Manager slider > Global settings.



2. Click System. The System window appears in the content area.
3. Scroll to the Management interface access list section and click **Add**.
4. **IP address**—Enter the IP address representing IP address of the destination network for which you want to allow over the management interface.
5. **Subnet mask**—Enter the netmask portion of the destination network for the traffic you want to allow.

The screenshot shows a dialog box titled 'Add Management interface access list'. It has two input fields: '*IP address' with the value '111.33.45.10' and a tooltip '(Unique required. Enter a valid IP address)', and '*Subnet mask' with the value '225.225.0.0' and a tooltip '(Enter a valid IP address)'. At the bottom are 'Apply' and 'Cancel' buttons.

6. Click **Apply**. The IP address and subnet mask appear in the Management interface access list.
7. Repeat steps 4 through 7 to add more IP addresses and subnet masks you want to allow as packet sources.
8. Click **Apply**.

Editing Management Interface Access Lists

1. Click a management interface access list in the Management interface access list table and click **Edit**.
2. **Subnet mask**—Edit the netmask portion of the destination network for the traffic you want to allow.

3. **Description**—Enter an optional descriptive string to identify the management interface access list.

*IP address	100.10.20.40
*Subnet mask	111.33.15.20
Description	Enter description here
Protocol	Add Delete

4. Click **Apply**.

Notes on Deleting System ACLs

If you delete a system ACL from your configuration, the Net-Net SBC checks whether or not there are any active FTP or Telnet client was granted access when the entry was being removed. If such a client were active during ACL removal, the Net-Net SBC would warn you about the condition and ask you to confirm the deletion. If you confirm the deletion, then the Net-Net SBC's session with the active client is suspended.

IPSec Manual Keying

Net-Net SBC supports IPSec manual keying. Manual keying can be used in the creation of an IPSec tunnel to another endpoint that does not support IKE, but does support IPSec. With IPSec manual keying, you can enter the shared secret key required in hashing or encrypting for both sides of the data flow.

Supported Protocols

The Net-Net SBC's IPSec implementation supports all required tools for securing Internet communication via the IPSec protocol suite. The following paragraphs list and explain the protocols within the IPSec suite that the Net-Net SBC supports. This chapter does not explain how to design and choose the best protocols for your application.

AH vs. ESP

The Net-Net SBC supports the two encapsulations that IPSec uses to secure packet flows. Authentication Header (AH) is used to authenticate and validate IP packets. Authentication means that the packet was sent by the source who is assumed to have sent it. Note that AH is incompatible with NAT. Validation means that the recipient is assured that the packet has arrived containing the original, unaltered data as sent.

ESP (Encapsulating Security Payload) provides AH's authentication and validations and extends the feature set by ensuring that the IP packet's contents remain confidential as they travel across the network. Using an encryption algorithm that both peers agree upon, ESP encrypts a full IP packet so that if intercepted, an unauthorized party cannot read the IPsec packet's contents.

Tunnel Mode vs. Transport Mode

In addition to its security encapsulations, the IPsec suite supports two modes: tunnel mode and transport mode. Tunnel mode is used most often for connections between gateways, or between a host and a gateway. Tunnel mode creates a VPN-like path between the two gateways and encapsulates the entire original packet. Transport mode is used to protect end-to-end communications between two hosts providing a secured IP connection and encrypts just the original payload.

Cryptographic Algorithms

IPsec works by using a symmetric key for validation and encryption. Symmetric key algorithms use the same shared secret key for encoding and decoding data on both sides of the IPsec flow. The Net-Net SBC's IPsec feature supports the following encryption algorithms:

- DES
- 3DES
- AES128CBC
- AES256CBC
- AES128CTR
- AES256CTR

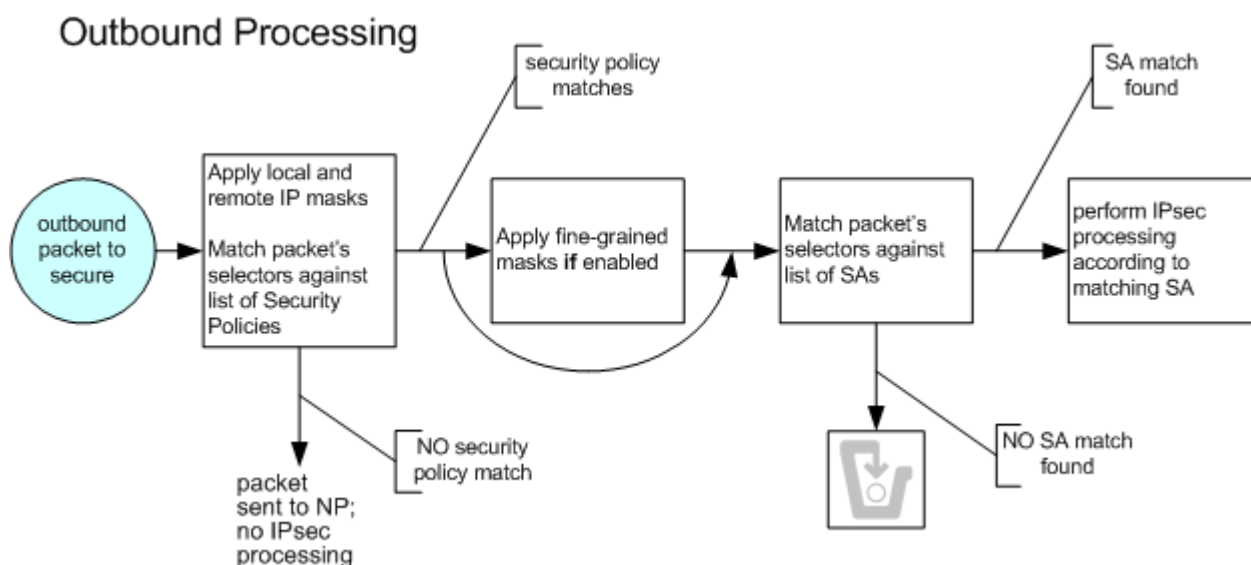
The Net-Net SBC can quickly generate keys for all of the above mentioned algorithms from the CLI. It can additionally supports HMAC-SHA1 or HMAC-MD5 keyed-hash message authentication codes. Only manual keying is currently supported for both hash authentication and data encryption. Therefore, all keys must be provisioned on the Net-Net SBC by hand.

IPsec Implementation

The Net-Net SBC uses separate logic for processing IPsec packets based on whether the traffic is inbound or outbound. The configuration is divided into two pieces, the security policy and the security association (SA). Both the SA and security policies have a directional attribute which indicates if they can be used and/or reused for inbound and outbound traffic.

Outbound Packet Processing

The following diagram shows the steps the Net-Net SBC follows when processing outbound IPsec traffic. Details of each step are described in the following sections.



Security Policy

The Net-Net SBC first performs a policy lookup on outbound traffic to test if it should be subjected to IPsec rules. A security policy, local policy only applicable for IPsec functionality, defines the matching criteria for outgoing network traffic to secure. It is configured on a network interface basis.

Configuring a security policy is similar to a local policy, with additional IPsec-specific parameters. Unlike a local policy, used for routing, a security policy is used for packet treatment. As with any local policy, a set of selector values is matched against the outbound flow's following characteristics:

- VLAN
- Source IP address (plus mask)
- Source IP port
- Destination IP address (plus mask)
- Destination IP port
- Transport Protocol

Each of these selection criteria can be defined by a wildcard except for the VLAN ID, which can be ignored. This flexibility aids in creating selection criteria that ranges from highly restrictive to completely permissive. In addition to the main traffic matching criteria, a priority parameter is used to prioritize the order that configured security policies are checked against. The #0 policy is checked first, #1 policy is checked next, continuing to the lowest prioritized policy being checked last.

Once the outbound traffic matches a policy, the Net-Net SBC proceeds to the next step of outbound IPsec processing. If no matching security policy is found, the default pass-through policy allows the packet to be assembled and forwarded to the network without any security processing.

Fine-Grained Policy Selection

After a positive match between outbound traffic and the configured selectors in the security policy, the Net-Net SBC can perform a calculation between a set of fine-

grained packet selectors and the outbound packet. The fine-grained policy masking criteria are:

- Source IP subnet mask
- Destination IP subnet mask
- VLAN mask

By default, the fine-grained security policy is set to match and pass all traffic untouched to the security association (SA) portion of IPsec processing.

Fine-grained policy selection works by performing a logical AND between outbound traffic's fine-grained selectors and the traffic's corresponding attributes. The result is then used to find the matching SA. Applying a fine-grained mask has the effect of forcing a contiguous block of IP addresses and/or ports to appear as one address and or port. During the next step of IPsec processing, when an SA is chosen, the Net-Net SBC in effect uses one SA lookup for a series of addresses. Without fine-grained policy selection, unique SAs must always be configured for outbound packets with unique security policy selectors.

Security Associations

After the Net-Net SBC determines that outgoing traffic is subject to IPsec processing, and optionally applies fine-grained masking, an SA lookup is performed on the traffic. An SA is the set of rules that define the association between two endpoints or entities that create the secured communication. To choose an SA, the Net-Net SBC searches for a match against the outgoing traffic's SA selectors. SA selectors are as follows:

- VLAN
- Source IP address
- Source IP port
- Destination IP address
- Destination IP port
- Transport protocol

If there is a match, the Net-Net SBC secures the flow according to security parameters defined in the SA that the Net-Net SBC chooses. The packet is then forwarded out of the Net-Net SBC. If no match is found, the packets are discarded, and optionally dumped to secured.log if the log-level is set to DEBUG.

Secure Connection Details

Several parameters define an IPsec connection between the Net-Net SBC and a peer. When planning an IPsec deployment, the primary architectural decisions are which IPsec protocol and mode to use. The two choices for IPsec protocol are ESP or AH, and the two choices for IPsec mode are either tunnel or transport. IPsec protocol

and mode are both required for an SA configuration. When creating an IPsec tunnel (tunnel mode), the SA must also define the two outside IP addresses of the tunnel.

Remote tunnel IP address

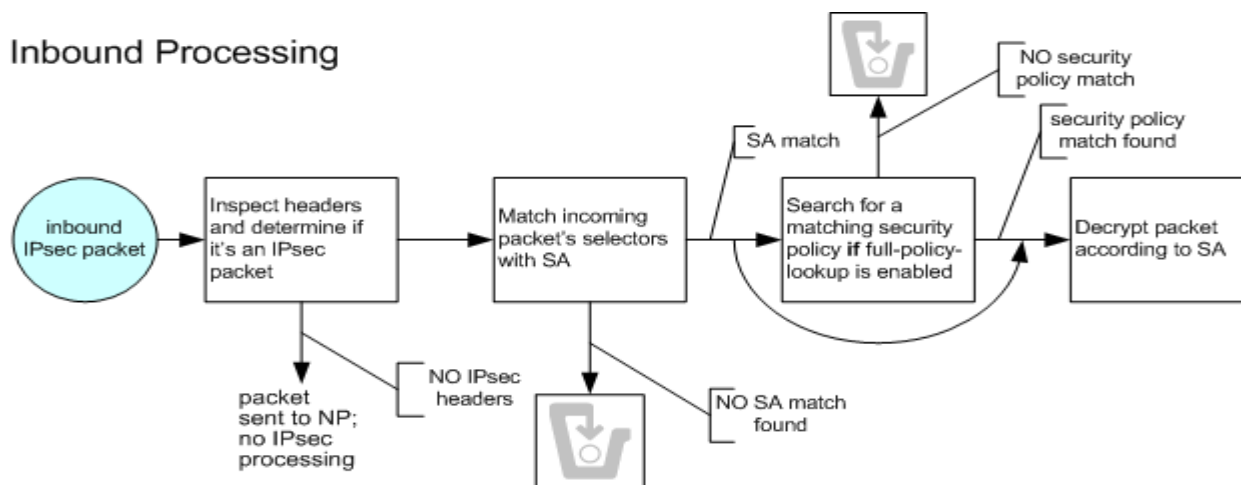


The authentication algorithm and the authentication key are always configured. The Net-Net SBC supports hmac-md5 or hmac-sha1 authentication algorithms. Because only manual keying is supported, the key must be entered by hand. When encryption is required, the encryption algorithm and the encryption key must be configured. The Net-Net SBC supports des, 3des, aes-128-cbc, aes-256-cbc, aes-128-ctr, and aes-256-ctr encryption algorithms. When using the two encryption protocols that operate in counter mode (RFC 3686), an additional nonce value is required. In addition, the security parameter index (SPI) must be configured for each SA.

Inbound Packet Processing

The following diagrams shows the steps the Net-Net SBC follows when processing inbound IPsec traffic. Details of each step are described in the following sections.

Inbound Processing



IP Header Inspection

Processing inbound IPsec packets begins by the Net-Net SBC inspecting an inbound IP packet's headers. If the packet is identified as IPsec traffic, an SA policy lookup is performed. If the traffic is identified as non-IPsec traffic, it is forwarded to the NP without any security processing.

SA Matching

The Net-Net SBC proceeds to match the inbound IPsec traffic's selectors against configured SAs. Inbound selector masking is performed where noted. These selectors are:

- VLAN (plus mask)

- Source IP address (plus mask)
- Source IP port
- Destination IP address (plus mask)
- Destination IP port
- Transport Protocol
- SPI

If no matching SA is found, the packets are discarded, and optionally dumped to secured.log if the log-level is set to DEBUG. When the Net-Net SBC finds a matching SA, the packet is authenticated and decrypted according to the configuration and sent to the Net-Net SBC's NP for continued processing.

Inbound Full Policy Lookup

Inbound traffic can optionally be subjected to a full policy lookup, prior to decryption and authentication. A full policy lookup checks if a security policy exists for this inbound traffic before the Net-Net SBC proceeds to decrypt and authenticate it. If no matching security policy is found, even after a successful SA match, the packets are discarded, and optionally dumped to secured.log if the log-level is set to DEBUG. Full policy lookups consume additional processing, and should not be configured unless absolutely useful.

HA Considerations

Anti-replay mechanisms, running on IPsec peers, can cause instability with the Net-Net SBCs configured in an HA pair. The anti-replay mechanism ensures that traffic with inconsistent (non-incrementing) sequence numbers is labeled as insecure, assuming it could be part of a replay attack. Under normal circumstances, this signature causes the remote endpoint to drop IPsec traffic.

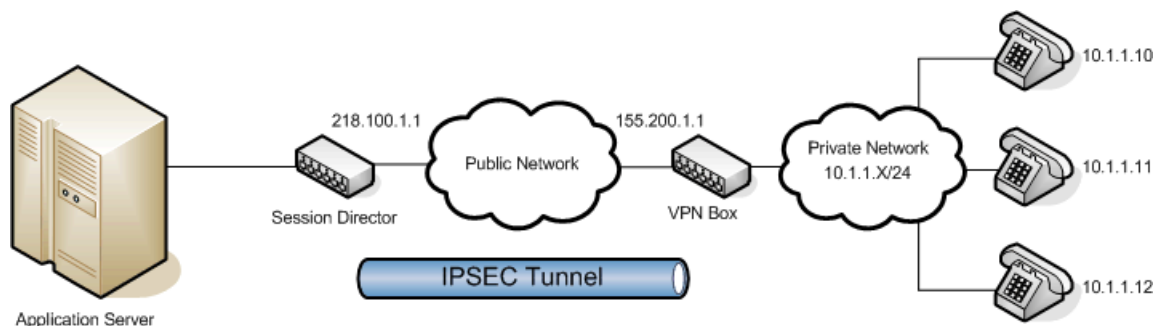
When a failover occurs between HA peers, the newly-active system starts sending traffic with the IPsec sequence number starting at 0. A remote system's anti-replay mechanism observes this and labels the traffic as insecure. It is therefore recommended that anti-replay protection not be used with Net-Net SBCs in an HA configuration. This situation does not create any problems as long as IPsec peers are not configured to use anti-replay mechanisms.

Packet Size Considerations

The security processor supports receipt of jumbo frames up to 9K (9022 bytes with VLANs, 9018 without). Under normal operation the default outgoing maximum packet size of 1500 bytes is used. This packet size includes the IPSEC headers, which will result in less space for packet data (SIP signaling, RTP, etc...).

IPsec Application Example

In this example, the Net-Net SBC terminates an IPsec tunnel. The remote side of the tunnels is a dedicated VPN appliance in the public Internet. Behind that VPN appliance are three non-IPsec VoIP phones. In this scenario, the VPN box maintains the IPsec tunnel through which the phones communicate with the Net-Net SBC.



Without the fine-grained option (or alternatively IKE), an SA entry would need to be configured for each of the three phones, communicating over the IPSEC tunnel (resulting in 3 tunnels).

This does not scale for manual-keying with a large number of endpoints. Using the fine-grained configuration as well as the inbound SA mask allows any number of endpoints on the 10.1.1.X network to use a single security association (a coarse-grain configuration). The configuration in this example follows a packet sent from the Net-Net SBC to any of the phones will match the policy pol1.

The remote IP mask value of the fine-grained configuration will then be masked against the remote IP, resulting in a SA selector value of 10.1.1.0. This matches security association sa1, and the packet will be secured and sent over the tunnel. The tunnel mode addresses in the security association represent the external, public addresses of the termination points for the IPSEC tunnel.

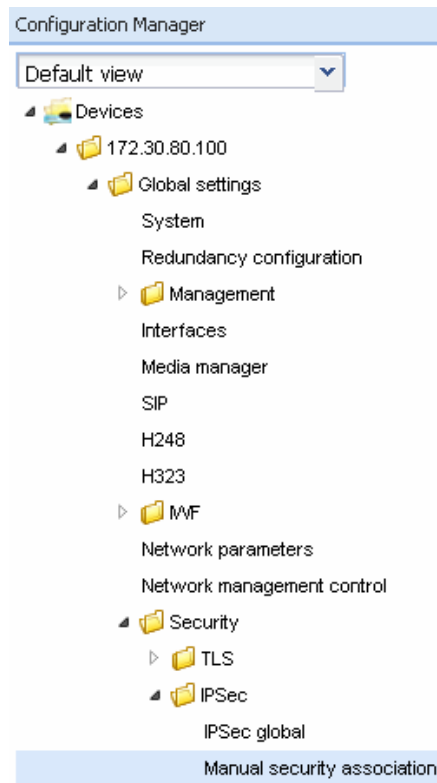
Packets returning from the 10.1.1.0 side of the IPsec tunnel will first match the tunnel mode local IP addr of 218.100.1.1. The packets will then be decrypted using the SA parameters, and the tunneled packet will be checked against the remote IP address value of the SA.

If the fine-grained mask had not been used, three discrete SAs would have to be configured: one for each of the three phones.

Configuring Security Associations

To configure security associations:

1. Expand the Configuration Manager slider > Global settings > Security > IPSec.



2. Click Manual security association.
3. Click **Add**. The Add manual security association dialog box appears.
4. **Name**—Enter a name for the security association.
5. **Local IP address**—Enter the local IP address to match for traffic selectors for this SA.
6. **SPI**—Set the security parameter index.
 - minimum: 256
 - maximum: 2302

The default value is 256.
7. **Remote IP Address**—Enter the remote IP address to match for the traffic selectors for this SA.
8. **Sub port**—Enter the identification of a specific virtual interface in a physical interface. A value of 0 indicates that this element is not using a virtual interface.

9. **Interface name**—Select the network interface where this association applies.

Add Manual security association

*Name: SA1 (Unique required)

*Local IP address: 100.10.20.30 (Enter a valid IP address. Default:0.0.0.0)

*SPI: 256 (Range:256..2302. Default:256)

*Remote IP address: 0.0.0.0 (Enter a valid IP address. Default:0.0.0.0)

Network interface

*Sub port: 111 (Default:0)

*Interface name: M00

Configuration Guidelines

Apply Cancel

10. Click **Apply**. The security association now appears in the table.
11. Select the security association from the table and click **Edit**.
12. **IPSec mode**—Choose the mode from the combination box.
- If you choose tunnel mode, you need to edit this security association to configure the tunnel mode. See *Configuring Tunnel Mode* for details.
13. **IPsec protocol**—Choose the protocol from the combination box.
14. **Transport protocol**—Choose the protocol from the combination box.
15. **Direction**—Choose the direction of traffic this security policy can apply to from the combination box.
- in—valid for inbound traffic
 - out—valid for outbound traffic
 - both—valid for inbound and outbound traffic
16. **Local IP address**—Enter the the local IP address for the security association to match.
17. **Local port**—Enter the port value to match in the range 0 to 65535.
18. **Remote IP address**—Enter the remote endpoint IP address to match.
19. **Remote port**—Enter the remote endpoint port number to match.

IPSec protocol	esp
Transport protocol	ALL
Direction	both
*Local IP address	100.10.20.30
Local port	0
*Remote IP address	0.0.0.0
Remote port	0

20. **Authentication algorithm**—Choose the algorithm from the combination box.

Authentication algorithm

hmac-sha1

none

hmac-sha1

hmac-md5

21. **Authentication key**—Click **Edit**.
22. **New value**—Enter the 16 byte hexadecimal number for hmac-md5 or the 20 byte hexadecimal number for hmac-sha1.
23. **Confirm new value**—Retype the new value to confirm the value.
24. **Configuration password**—Enter the password used to encrypt the configuration information on the Net-Net SBC.
25. **Confirm configuration password**—Retype the password to confirm the password.

Edit encryption password.

Enter new value.

Confirm new value.

Enter configuration password.

Confirm new value.

OK Cancel

26. Click **OK**. A message appears recommending you run the ACLI command `verify-config` on the Net-Net SBC after you save your configuration

Info

It is recommended that you run `verify-config` on SBC after saving this config to the SBC.

OK

27. Click **OK** to clear the message. The asterisks that represent the authentication key appear in the list.
28. **Encryption algorithm**—Choose the algorithm from the combination box.

Encryption algorithm

aes-256-cbc

3des

none

aes-256-cbc

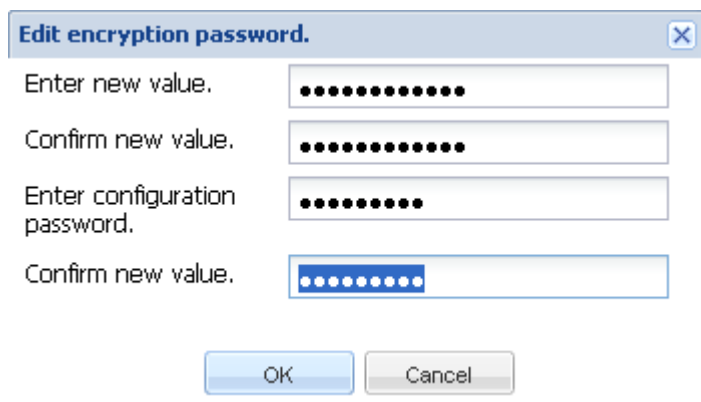
aes-128-ctr

aes-256-ctr

aes-128-cbc

29. **Encryption key**—Click **Edit**. The Edit security dialog box appears.

30. **New value**—Enter the string you want to use as the key value.
31. **Confirm new value**—Retype the new value to confirm the value.
32. **Configuration password**—Enter the password used to encrypt the configuration information on the Net-Net SBC.
33. **Confirm configuration password**—Retype the password to confirm the password.



The dialog box titled "Edit encryption password." contains four input fields, each with a label and a text box filled with asterisks. The labels are "Enter new value.", "Confirm new value.", "Enter configuration password.", and "Confirm new value.". At the bottom of the dialog are "OK" and "Cancel" buttons.

34. Click **OK**. A message appears recommending you run the ACLI command `verify-config` on the Net-Net SBC after you save your configuration



The "Info" dialog box features a yellow warning icon and the text: "It is recommended that you run verify-config on SBC after saving this config to the SBC." Below the text is an "OK" button.

35. Click **OK** to clear the message. The asterisks that represent the encryption key appear in the list.
36. **AES counter mode nonce**—Enter the nonce for AES in the counter mode if you chose `aes-128-ctr` or `aes-256-ctr` as your encryption algorithm.

AES counter mode nonce

You need to edit this security association to configure the network interface. If you chose tunnel as the IPSec mode, you need to edit the security association to configure the tunnel mode.

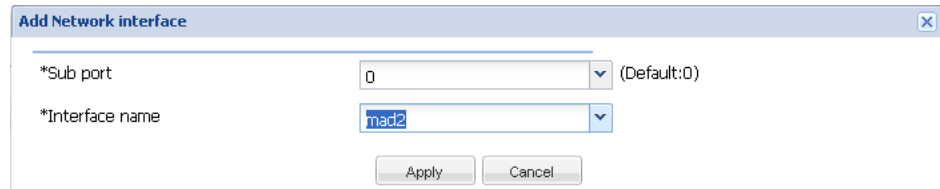
Configuring Network Interfaces

You need to edit the security association to configure the network interface.

To configure network interfaces:

1. In the Manual security association table—click the security association you want to edit and click **Edit**.
2. **Network interface**—Click **Add**. The Add network interfaces dialog box appears.
3. **Sub port**—Choose the subport from the combination box.

4. **Interface name**—Choose the interface from the combination box.



The 'Add Network interface' dialog box contains two fields: '*Sub port' with a value of '0' and '(Default:0)', and '*Interface name' with a dropdown menu showing 'mad2'. At the bottom are 'Apply' and 'Cancel' buttons.

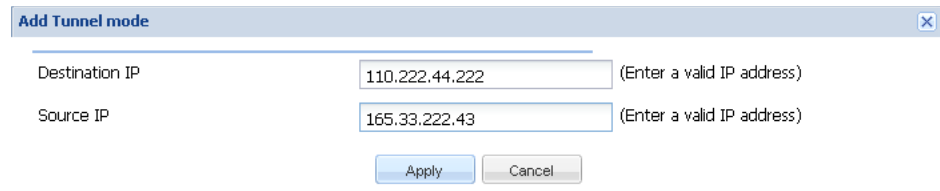
5. Click **Apply**.
6. Click **Apply**.

Configuring Tunnel Mode

To configure tunnel mode:

If you choose tunnel as the IPSec mode, you need to configure the tunnel mode.

1. In the Manual security association table, click the security association you want to edit and click **Edit**.
2. In the Tunnel mode section, click **Add**. The Add Tunnel mode dialog box appears.
3. **Source IP**—Enter the IP address for the source. Enter the local public IP address that terminates the IPSec tunnel.
4. **Destination IP**—Enter the destination's public IP address that terminates the IPSec tunnel.



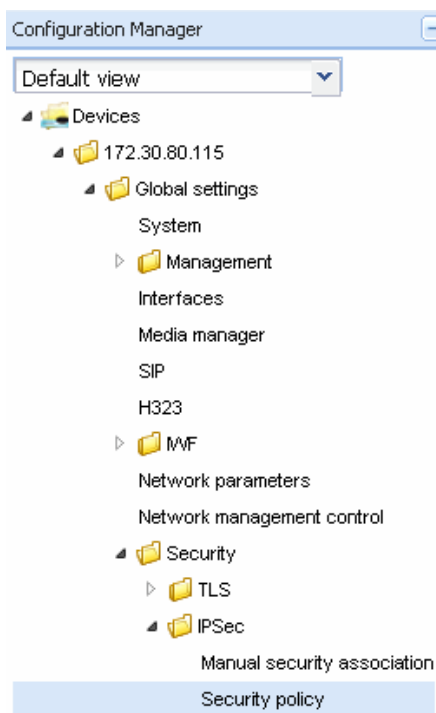
The 'Add Tunnel mode' dialog box contains two fields: 'Destination IP' with the value '110.222.44.222' and '(Enter a valid IP address)', and 'Source IP' with the value '165.33.222.43' and '(Enter a valid IP address)'. At the bottom are 'Apply' and 'Cancel' buttons.

5. Click **Apply**.
6. Click **Apply**.

Configuring Security Policies

To configure security policies:

1. Expand the Configuration Manager slider > Global settings > Security > IPSec.



2. Click Security policy.
3. Click **Add**. The Add Security policy dialog box appears.
4. **Name**—Enter the name for this security policy. This parameter is required.
5. **Priority**—Enter the priority number of this security policy. Valid values are 0 to 126.
6. **IP address match**—Enter the source IP address to match.
7. **IP address match**—Enter the destination IP address to match.
8. **Sub port**—Choose the subport from the combination box.

9. **Interface name**—Choose the interface from the combination box
Remote port match—Enter the destination port to match. A value of 0 disables this selector.

Add Security policy

*Name	secPolicy	(Unique required)
*IP address match	110.20.10.30	(Enter a valid IP address. Default:0.0.0.0)
*IP address match	220.10.30.40	(Enter a valid IP address. Default:0.0.0.0)
*Priority	0	(Range:0..126. Default:0)

[Configuration Guidelines](#)

Network interface

*Sub port	0	(Default:0)
*Interface name	mad2	

10. Click Apply. The new security policy appears in the table.
11. Select the security policy from the table and click **Edit**.
12. Scroll to Local.
13. **IP Mask**—Enter the source port to match. A value of 0 disables this selector.
14. **Port match**—Enter the source port to match. A value of 0 disables this selector.
15. Scroll to Remote.
16. **IP Mask**—Enter the remote port to match. A value of 0 disables this selector.
17. **Port match**—Enter the remote port to match. A value of 0 disables this selector.
18. **Transport protocol**—Select the transport protocol to match from the combination box.
19. **Action**—Select the action the Net-Net SBC should take when this policy matches outbound IPsec traffic from the combination box.
- **discard**—discard the traffic
 - **allow**—forward the traffic without any security processing
 - **ipsec**—continue processing as IPsec traffic
20. **Direction**—Select the direction of traffic this security policy can apply to from the combination box.
- **in**—valid for inbound traffic
 - **out**—valid for outbound traffic
 - **both**—inbound and outbound traffic

21. Click **Apply**.

Local

*IP address match	0.0.0.1
IP mask	255.255.255.255
Port match	0

Remote

*IP address match	0.0.0.2
IP mask	255.255.255.255
Port match	0

Transport protocol	all ▼
Action	ipsec ▼
Direction	both ▼

Configuring Outbound Fine-Grained SA Matching Criteria

To configure outbound fine-grained SA matching criteria:

By default, the fine-grained configuration is configured on a new policy with the local IP mask and remote IP mask parameters set to 255.255.255.255. This essentially leaves the SA selectors intact (unmasked) when performing the SA lookup, resulting in an exact match.

1. In the Security policy data screen, scroll to Outbound SA fine grained mask.
2. **Local IP mask**—Enter the the fine-grained source IP address mask to apply to outbound IP packets for SA matching.
3. **Remote IP mask**—Enter the fine-grained destination IP address mask to apply to outbound IP packets for SA matching.
4. **VLAN ID mask**—Enter the fine-grained VLAN mask to apply to outbound IP packets for SA matching. Valid values range from 0x000 through 0xFFF
5. **Local port mask**—Enter the local port mask for this security policy. Valid values range from 0 (minimum) to 65535 (maximum). The default value is 0.
6. **Remote port mask**—Enter the remote port mask for this security policy. Valid values range from 0 (minimum) to 65535 (maximum). The default value is 0.

7. **Transport protocol mask**—Enter the transport protocol mask for this security policy. The valid values range from 0 (minimum) to 255 (maximum). The default value is 0..

Add Outbound SA fine grained mask		
Transport protocol mask	45	(Range: 0..255, Default: 0)
Local port mask	10	(Range: 0..65535, Default: 0)
Remote port mask	12	(Range: 0..65535, Default: 0)
Remote IP mask	255.255.255.255	(Enter a valid IP address, Default: 255.255.255.255)
VLAN ID mask	4095	(Range: 0..4095, Default: 4095)
Local IP mask	255.255.255.255	(Enter a valid IP address, Default: 255.255.255.255)
Apply Cancel		

8. Click **Apply**.

National Security and Emergency Preparedness for SIP

The Net-Net SBC supports the Emergency Telecommunications Service (ETS), which gives priority treatment of National Security and Emergency Preparedness (NSEP) communications for IP network infrastructures. ETS can increase the likelihood that calls, sessions, and other communications will be successfully completed when they are initiated by government-authorized users over the public network infrastructure. Legacy circuit-switched services such as Government Emergency Telecommunications Service (GETS) and Wireless Priority Service (WPS) also fall under the ETS rubric, and are now also supported on the Net-Net SBC.

To provide this support, you can enable the Net-Net SBC to act on SIP calls that contain an ETS dial number (DN) and/or the SIP Resource-Priority header that carries ETS resource values.

How It Works

The Net-Net SBC identifies ETS calls by using the system's pre-existing network management controls (NMC) functionality. With NMC and Resource-Priority header (RPH) support enabled on your system, the Net-Net SBC detects an ETS calls and provides the appropriate treatment for them.

The Net-Net SBC supports this feature by treating ETS calls based on the r-value parameter in the Resource-Priority header. The r-value is a key piece of information because it defines the resource priority that the call originator requests. The r-value parameter provides namespaces and priorities that the Net-Net SBC can manipulate in outgoing traffic.

The RPH profile is applied to an NMC rule, where they determine r-values, a media policy to use, and what type of call treatment to apply. Also applies to an NMC rule, the new RPH policy configuration provides information about which r-values to insert and which to override.

Licensing

To enable NSEP for SIP on your Net-Net SBC, you must obtain and install a new license. For information about how to obtain an NSEP RPH license, contact your Acme Packet sales representative.

Matching by NMC and by RPH

When a Net-Net SBC has been enabled to act on RPH, it checks incoming requests for RPH, tries to parse that RPH, and then rejects requests in the circumstances listed below. For all of these rejections, the Net-Net SBC logs the error at the TRACE level.

- Request with multiple instances of the same namespace in the RPH—The Net-Net SBC sends out a **400 Bad Request** response with the “Invalid RPH - Namespace repeated” header showing that there are multiple instances of the same namespace in the RPH.
- Request with invalid resource priority for a namespace—The Net-Net SBC sends out a **400 Bad Request** response with the “Invalid RPH - Invalid rvalue: x” showing that there is an invalid resource value (where x is the invalid value).
- Request with WPS namespace, but without ETS namespace—The Net-Net SBC sends out a **400 Bad Request** response with the “Invalid RPH - No ETS value” header showing that there is no ETS namespace.

If the Net-Net SBC successfully parses the RPH, it identifies the ETS call by checking the Request-URI of the incoming request against destination identifiers that you configure in the NMC rules. If there is a match between the request’s ETS DN and the destination value identifier in the NMC rules, the Net-Net SBC tags the call; note that NMC rules need to be configured with the `rph-feature` parameter set to enabled to identify an ETS call properly. If there is no match to an NMC rule, then the Net-Net SBC performs matching based on RPH by comparing resource values (r-values) in the RPH with values you set in the RPH profile configuration.

For an ETS call that matches by ETS DN and NMC rule, the Net-Net SBC checks the NMC rule to determine if it has an RPH profile (with r-values) assigned to it. If so, the Net-Net SBC continues by comparing the RPH profile’s r-values against those in the request’s RPH. In cases where the RPH does not contain a recognized value r-value, the Net-Net SBC:

- Processes the call as it normally would (as a non-ETS call) without changing the RPH if the resource-priority option tag is not present in the Required header (for an INVITE only and not any other requests or response from which RPH would be deleted)
- Rejects the Request when the Require header has the resource-priority header; or, inserts an Accept-Resource-Priority header (ARPH) in the response if the **insert-arp-header** parameter option is enabled

However, the call goes through the Net-Net SBC as an ETS call when it is matched by ETS DN and the applicable NMC does not have an RPH profile assigned. According to the settings in the NMC rule, the Net-Net SBC either diverts or rejects such a call. And when the call matches by RPH rather than ETS DN, the Net-Net SBC applies the configured RPH profile from the relevant NMC rule.

It can be the case that non-ETS calls have RPH in their requests. Here, the Net-Net SBC call treatment is performed according to the settings in the matching RPH profile when there is no matching NMC rule. When you configure treatment as “reject,” then the Net-Net SBC rejects the call with a **417 Unknown-Resource Priority** status code. When you set the treatment to either “accept” or “priority,” the Net-Net SBC allows the call to proceed as a non-ETS call or as a priority call.

The ETS r-value can appear in ACK, BYE, INFO, PRACK, REFER and UPDATE requests. In cases when it does and the session with which the request is associated is a non-ETS call, the Net-Net SBC removes the RPH from the request before forwarding it and logs a TRACE-level error. The Net-Net SBC also removes RPH from responses before forwarding them and logs a TRACE-level error when responses contain RPH headers with ETS values for non-ETS sessions.

Call Treatment

This section describes how ETS calls are treated as they traverse the Net-Net SBC.

Call Treatment	Description
Routing	ETS calls are routed the same way as any other calls are, except when the applicable NMC rule's treatment type is "divert," and rule defines the next hop. This route takes precedence over other normal routes.
Local NMC	ETS calls are exempt from the local NMC, including: session agent constraints, bandwidth constraints (e.g., per-realm bandwidth), per-user CAC, and CPU constraints. However, the call is subject to the ETS congestions control threshold. Licensing session constraints apply.
ETS Call Congestion Control	ETS calls are subject to congestion control constraints that you configure specifically for this type of traffic. In the global SIP configuration, you set up one option that defines a load limit (greater than that set for normal calls).
ETS CAC	Although the Net-Net SBC uses the call rate control value in the applicable NMC rule, you can also enforce call rate on a per-user basis for ETS calls.

Generating Egress RPH

For each ETS call, the Net-Net SBC generates RPH for the outgoing request. It forms this RPH according to the information in the NMC rule. The outgoing request types are INVITE, ACL, BYE, CANCEL, INFO, PRACK, REFER, and UPDATE.

Request RPH Status	Generated Egress RPH
Incoming request without RPH (matched by ETS DN)	Outgoing RPH value becomes the r-value set in the insert-r-value parameter in the RPH policy applied to the NMC rule.
Incoming request without RPH (matched by ETS DN)	If the insert-r-value parameter is empty in the RPH policy applied to the NMC rule or there is no RPH policy applied to the NMC rule, then the egress RPH will also not have RPH.
Incoming request has RPH	Egress RPH is the same as the ingress if the NMC rule has an RPH policy applied but the override-r-value for the policy is empty or if there is not RPH policy applied to the NMC rule. If the override-r-value for the policy is set, then the egress RPH is set to that value.

For example, given an incoming request with the resource priority `ets.0, dsn.flash` and an RPH policy with an override value of `wps.1, ets.1`, the egress request would be sent with a resource-priority of `wps.1, ets.1, dsn.flash`.

The Net-Net SBC also includes RPH in the following series of responses, even when the downstream SIP entity does not respond with an RPH: 1xx, 2xx, 3xx, 4xx, 5xx, and 6xx. The 401 `unauthorized` response is an exception.

Media Treatment

If the RPH profile set in an NMC names a media policy, then the Net-Net SBC implements it for the ETS call. This media policy overrides any media policy set in the realm configuration.

The possible Differentiated Services Code Point (DSCP) values for an ETS call are:

- Audio—Applied to the respective media for an ETS call
- Video—Applied to the respective media for an ETS call
- SIP—Applied to the ETS calls' SIP signaling messages, only for the egress call leg for the ETS session

Configuring ETS

This section shows you how to configure RPH profiles and policies that enable the Net-Net SBC to act on SIP calls that have an ETS DN and/or an RPH carrying ETS resources values. There are also settings for the global SIP configuration and for the NMC rule configuration that support this feature.

Valid values for the parameters that take r-values are `wps.x` and `ets.x`, where x is 0 through 4.

Setting Up an RPH Policy

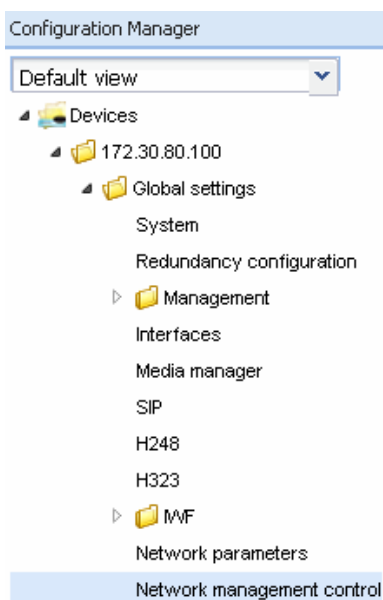
The RPH policy is a configuration on the Net-Net SBC that you apply to NMC rules. It designates the following for ETS/WPS namespaces:

- An override resource value—Resource value used to override the incoming RPH's resource value
- An insert resource value—Resource value inserted when the Net-Net SBC does not recognize the RPH, the incoming request has no RPH, or the call is H.323 and matches an NMC rule based on the ETS DN

Note that RPH policies do not apply for DSN, DRSN, Q.735, or any other type of namespace; these remain untouched in outgoing requests.

To configure an RPH policy:

1. Expand the Configuration Manager slider > Global settings.



2. Expand the Global settings folder and click Network management control.
3. Click the RPH policy tab at the top of the content area.

4. Click **Add**. The Add RPH policy dialog box appears.
5. **Name**—Enter a name that uniquely identifies the RPH Policy. In addition, this is the value you use to identify this policy for application in the NMC rules configuration.

6. Click **Apply**. The RPH policy appears in the RPH policy table.
7. Select the RPH policy from the table and click **Edit** to add r-values to the policy.
8. To add a value that the Net-Net SBC uses to override r-values in the original RPH, click **Add** in the right of the Override r-values table.
9. **R-value**—Enter the value you want to use.

10. Click **Apply**. The value you entered appears in the Override r-value table.
11. To add an value that the Net-Net SBC uses to insert r-values in the original RPH, click **Add** in the Insert r-value table.
12. **R-value**—Enter the value you want to use.
13. Click **Apply**. The value you entered appears in the Insert r-value table.
14. Click **Apply**. Your r-value entries appear in the RPH policy table on the RPH policy tab.

Setting Up an RPH Profile

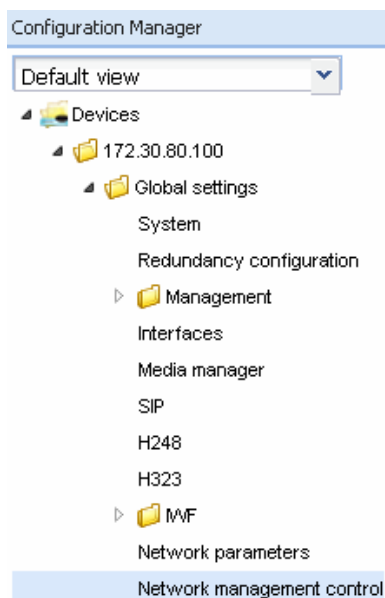
The RPH profile contains information about how the Net-Net SBC should act on the namespace(s) present in a Resource-Priority header (if any). The list of resource values in this configuration calls out the resource values (or r-values) recognizable to the Net-Net SBC; the ETS and WPS namespaces are supported.

You also set a media policy for the RPH profile to use; it defines the Differentiated Services Code Point (DSCP) that the Net-Net SBC uses for media or signaling packets belonging to the egress call leg for the ETS session.

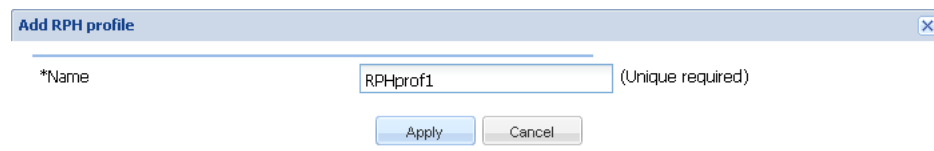
The call treatment parameter tells the Net-Net SBC what to do with a non-ETS call that has RPH in its request; the call can be allowed, rejected, or treated as a priority call.

To configure an RPH profile:

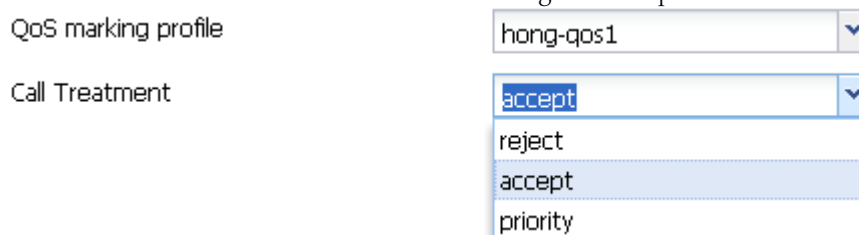
1. Expand the Configuration Manager slider > Global settings.



2. Expand the Global settings folder and click Network management control.
3. Click the RPH profile tab at the top of the content area. The RPH profile table appears.
4. Click **Add**. The Add RPH profile dialog box appears.
5. **Name**—Enter a name for this RPH Policy; this value uniquely identifies the RPH Policy. In addition, this is the value you use to apply the policy in the NMC rules configuration.



6. **QoS marking profile**—Select the QoS marking profile you want to apply for this RPH profile from the combination box. The Net-Net SBC implements this QoS marking profile for the ETS call, and it overrides any QoS marking profile set in the realm configuration.
7. **Call Treatment**—Choose a call treatment method from the combination box for a non-ETS call that contains RPH matching it to this profile.



The values are:

- **accept** (default, the call proceeds as it normally would)
- **reject** (the Net-Net SBC rejects the call with the 417 **Unknown-Resource Priority** status code)

- **priority** (Net-Net SBC treats the call as a priority call)
- In the Add RPH Profile window, click **OK** for the name, QoS marking profile, and call treatment to be applied. The values you entered appear in the RPH profile table on the RPH profile tab.
 - To Add r-values that the Net-Net SBC uses for matching purposes, highlight the RPH profile where you want to add the r-values and click **Edit**. The Edit RPH Profile window appears.
 - Next to the R-values table, click **Add**. The Add Override R-values window appears.
 - R-value**—Enter the r-value you want to use and click **Apply**.

You can add multiple r-values to a single RPH profile by separating the r-values in the entry by commas. You must enter them in the order reflected above (a WPS and then an ETS value). A WPS call always has to have an ETS namespace.

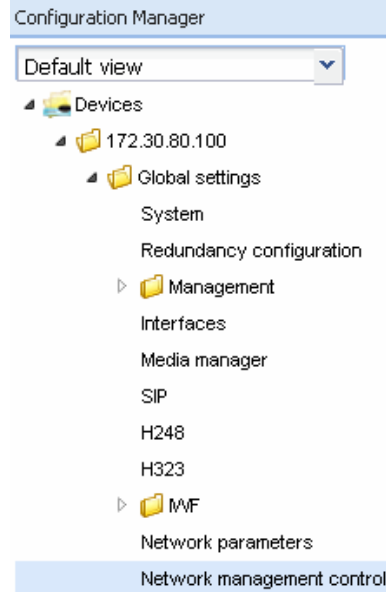
- When you are finished adding r-values, click **Apply** in the RPH profile window. The values you entered appear in the RPH profile table on the RPH profile tab.

Configuring an NMC Rule for NSEP

In addition to the RPH policy and RPH profile you can set for an NMC rule, you also need to set the state of this feature for the NMC rule.

To configure NMC rule for NSEP:

- Expand the Configuration Manager slider > Global settings.



- Expand the Global settings folder and click Network management control.
- Select a NMC rule that you want to configure for NSEP support from the table and click Edit.

4. **RPH enabled**—Click on the checkbox to enable the RPH/NSEP feature.

RPH enabled

enabled
 disabled

5. **RPH profile**—Select an RPH profile from the combination box.
6. **RPH policy**—Select an RPH policy from the combination box.
7. Click **Apply**.

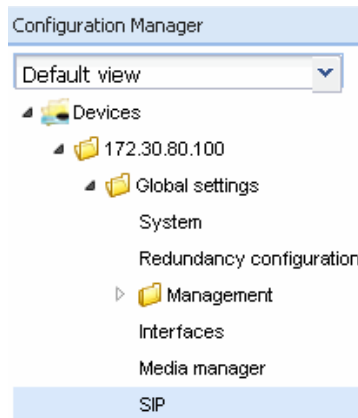
For the global SIP configuration, you can turn the NSEP feature on, and you can also set parameters that support call admission and congestion control.

In addition, you can enable the insertion of the ARPH header in a response when the resource-priority tag is present in the Require header and the Net-Net SBC rejects the request with a **417 Unknown Resource-Priority** response. The ARPH value is the list of r-values you set in the RPH profile.

Enabling NSEP and CAC

To enable NSEP support and call admission control (CAC):

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. **RPH enabled**—Select enabled from the combination box to enable RPH.
5. **NSEP user session rate**—To configure call admission, enter the maximum INVITEs per second to admit for ETS calls on a per-user basis for the NSEP user session rate. To enable NSEP CAC, you must change the parameter value from 0; if you leave this parameter set to 0, then it is the same as disabling CAC for ETS calls.

RPH enabled

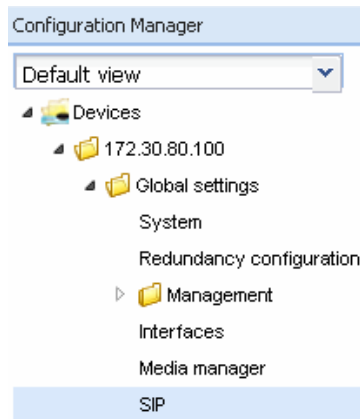
NSEP user sessions rate

6. Click **Apply**.

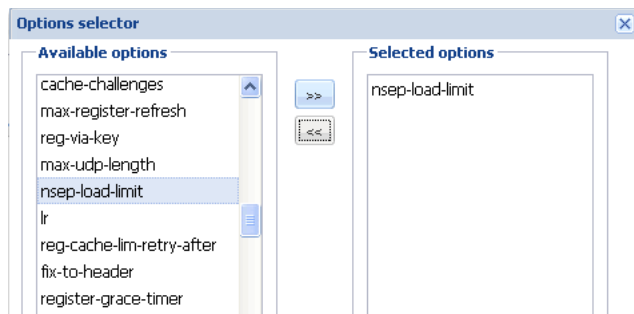
Enabling Congestion Control

To enable congestion control:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The SIP option window appears.
5. Click **nsep-load-limit** in the Available options list to select it.

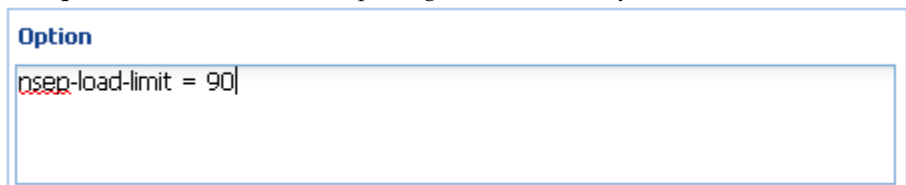


If the CPU threshold you set here is exceeded, the Net-Net SBC rejects new ETS calls with the **503 Service Unavailable** response.

6. Click  to move the option to the Selected list.

You need to specify a value for the load limit. The value you set here should be larger than the load limit value for normal calls; ETS calls are allowed even when the load limit threshold for normal calls is exceeded. The threshold value can be between 0 and 100. Using a value of 0 or 100 for this parameter disables ETS call congestion control.

7. **nsep-load-limit**—Enter an equal sign (=) followed by the threshold value.

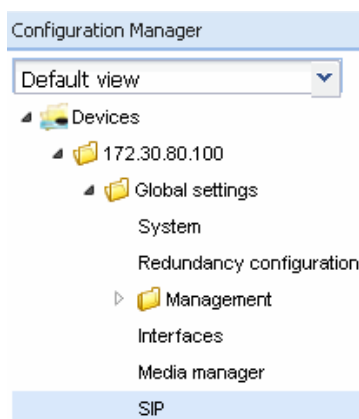


8. Click **OK**. The option with its threshold value appear in the Options section of the Options and tags window.

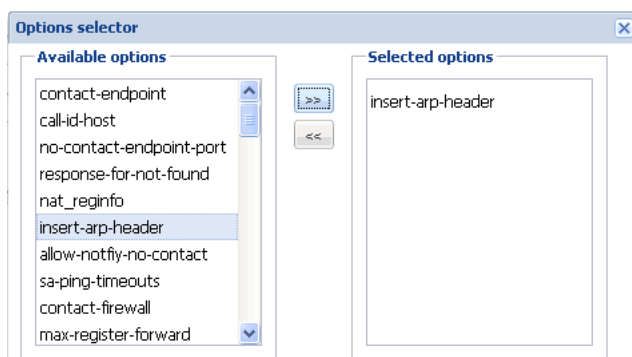
Enabling ARPH Insertion

To enable ARPH insertion in responses:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click **Options**. The SIP option window appears.
5. Click **insert-arp-header** in the Available options list to select it.



6. Click  to move the option to the Selected list.
7. Click OK to close the Options dialog box.

IDS Reporting

The Net-Net SBC supports a wide range of intrusion detection and protection capabilities for vulnerability and attack profiles identified to date. IDS reporting is useful for enterprise customers requirement to report on intrusions and suspicious behavior that it currently monitors.

IDS Licensing

This feature requires the IDS Reporting license. Note the following capabilities and restrictions of the license:

- The following configuration parameters are only visible after installing the license:
 - Trap on demote to deny
 - Syslog on demote to deny

- CAC failure threshold
- Untrusted CAC failure threshold
- Endpoint demotions based on admission control failures are only a valid option with the IDS License.
- The presence of the IDS license makes the apSysMgmtInetAddrWithReasonDOSTrap trap available and the apSysMgmtExpDOSTrap unavailable. Without an IDS licence installed, only the apSysMgmtExpDOSTrap trap is available.
- The **Trust->Untrust** and **Untrust-Deny** counters in the SIP and MGCP ACLs' statistics are visible regardless of the IDS license's presence.
- The **Demote Trust-Untrust** and **Demote Untrust-Deny** collect records in the SIP and MGCP ACL HDR groups are visible regardless of the IDS license's presence.
- A GET operation can be preformed on the two MIB entries to view the global endpoint counter for Demotions from Trusted to Untrusted and from Untrusted to Deny regardless of the IDS license's presence
- On Net-Net 3800 systems, the DOS license must be installed in addition to the IDS license in order to enable all features described in this section.

Basic Endpoint Demotion Behavior

Each session agent or endpoint is promoted or demoted among the trusted, untrusted, and denied queues depending on the **trust-level** parameter of the session agent or realm to which it belongs. Users can also configure access control rules to further classify signaling traffic so it can be promoted or demoted among trust queues as necessary.

An endpoint can be demoted in two cases:

- Net-Net SBC receiving too many signaling packets within the configured time window (**Maximum signal threshold in realms or access control**)
- Net-Net SBC receiving too many invalid signaling packets within the configured time window. (**Invalid signal threshold in realms or access control**)

Endpoint Demotion Reporting

The Net-Net SBC counts the number of endpoint or session agent promotions and demotions. Further, the Net-Net SBC counts when endpoints or session agents transition from trusted to untrusted and when endpoints transition from untrusted to denied queues. These counts are maintained for SIP and MGCP signaling applications.

SNMP Reporting

These per-endpoint counters are available under APSYSMGMT-MIB -> acmepacketMgmt -> apSystemManagementModule -> apSysMgmtMIBObjects -> apSysMgmtMIBGeneralObjects.

MIB NAME	MIB OID	PURPOSE
apSysSipEndptDemTrustToUnrust	.1.3.6.1.4.1.9148.3.2.1.1.19	Global counter for SIP endpoint demotions from trusted to untrusted.
apSysSipEndptDemUntrustToDeny	.1.3.6.1.4.1.9148.3.2.1.1.20	Global counter for SIP endpoint demotions from untrusted to denied.

MIB NAME	MIB OID	PURPOSE
apSysMgcpEndptDemTrustToUntrust	.1.3.6.1.4.1.9148.3.2.1.1.21	Global counter for MGCP endpoint demotions from trusted to untrusted.
apSysMgcpEndptDemUntrustToDeny	.1.3.6.1.4.1.9148.3.2.1.1.22	Global counter for MGCP endpoint demotions from untrusted to denied.

HDR Reporting

The SIP (sip-ACL-oper) and MGCP (mgcp-oper) HDR ACL status collection groups include the following two metrics:

- Demote Trust-Untrust (Global counter of endpoint demotion from trusted to untrusted queue)
- Demote Untrust-Deny (Global counter of endpoint demotion from untrusted to denied queue)

Endpoint Demotion SNMP Traps

An SNMP trap can be sent when the Net-Net SBC demotes an endpoint to the denied queue. This is set by enabling the **Trap on demote to deny** parameter located in the **media manager config** configuration element.

When the IDS license is installed and the **Trap on demote to deny** parameter is enabled, apSysMgmtInetAddrWithReasonDOSTrap trap is sent. This trap supersedes the apSysMgmtInetAddrDOSTrap trap.

When the IDS license is installed and the **Trap on demote to deny** parameter is disabled the apSysMgmtInetAddrWithReasonDOSTrap trap is not sent from the Net-Net SBC, even when an endpoint is demoted to the denied queue.

This apSysMgmtInetAddrWithReasonDOSTrap contains the following data:

- apSysMgmtDOSInetAddressType—Blocked IP address family (IPv4 or IPv6)
- apSysMgmtDOSInetAddress—Blocked IP address
- apSysMgmtDOSRealmID—Blocked Realm ID
- apSysMgmtDOSFromURI—The FROM header of the message that caused the block (If available)
- apSysMgmtDOSReason—The reason for demoting the endpoint to the denied queue: This field can report the following three values:
 - Too many errors
 - Too many messages
 - Too many admission control failures

Note: By default, this parameter is enabled for upgrade configurations, as the current behavior is to send a trap for every endpoint that is demoted to deny. However, for a new configuration created, the value to this configuration is disabled.

Endpoint Demotion Syslog Message

A Syslog message can be generated when an endpoint is demoted. Setting the **media manager config -> syslog-on-demote-to-deny** parameter to enabled writes and endpoint demotion warning to the syslog every time an endpoint is

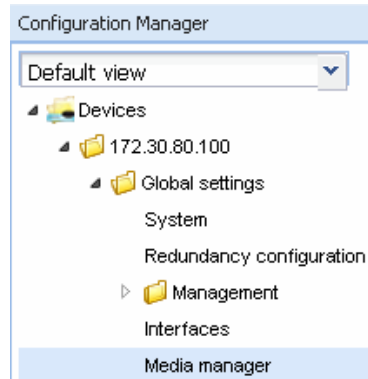
demoted to the denied queue. By default, this configuration option is set to disabled. The syslog message has a WARNING Level and looks like this:

```
Jan 15 12:22:48 172.30.60.12 ACMEsystem sipd[1c6e0b90] WARNING
SigAddr[access:168.192.24.40:0=low:DENY] ttl=3632 guard=798 exp=30
Demoted to Black-List (Too many admission control failures)
```

Configuring Endpoint Demotion

To configure the Net-Net SBC to send traps and/or write syslog messages on endpoint demotion:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. **Send ACL trap on demote to deny**—Select enabled from the combination box to enable the Net-Net SBC to send the apSysMgmtInetAddrWithReasonDOSTrap trap when applicable.
5. **Syslog on demote to deny**—Select enabled from the combination box to enable the Net-Net SBC to write an endpoint demotion warning message to the syslog.

Send ACL trap on demote To deny	enabled
Send syslog on demote To deny	enabled enabled disabled

6. Click **Apply**.

Endpoint Demotion due to CAC Overage

The Net-Net SBC can demote endpoints from trusted to untrusted queues when CAC failures exceed a configured threshold. The Net-Net SBC can also demote endpoints from untrusted to denied queues when CAC failures exceed a another configured threshold.

The Net-Net SBC maintains CAC failures per-endpoint. The CAC failure counter is incremented upon certain admission control failures only if either one of **CAC failure threshold** or **Untrusted CAC failure threshold** is non-zero.

The **CAC failure threshold** parameter is available in the access control and realm configuration elements. Exceeding this parameter demotes an endpoint from the trusted queue to the untrusted queue. The **Untrusted CAC failure threshold**

parameter is available in the access control and realm configuration elements. Exceeding this parameter demotes an endpoint from the untrusted queue to the denied queue.

If both the **CAC failure threshold** and **Untrusted CAC failure threshold** are configured to 0, then admission control failures are considered and counted as invalid signaling messages for determining if the **Malformed messages** parameter value has been exceeded.

CAC Attributes used for Endpoint Demotion

The Net-Net SBC determines CAC failures only by considering the calling endpoint's signaling messages traversing the calling realms' configuration. If an endpoint exceeds the following CAC thresholds, the Net-Net SBC will demote the endpoint when the CAC failure thresholds are enabled.

- SIP interface user CAC sessions
- SIP interface user CAC bandwidth
- External policy server rejects a session

Authentication Failures used for Endpoint Demotion

If an endpoint fails to authenticate with the Net-Net SBC using SIP HTTP digest authentication OR endpoint fails authentication with an INVITE with authentication incase registration-caching is disabled, and receives back a 401 or 407 response from the registrar

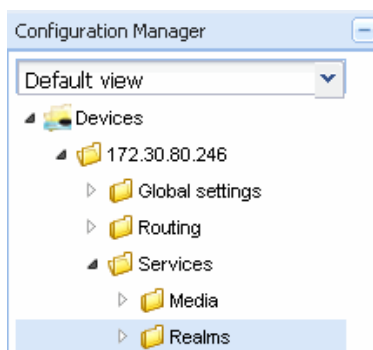
When the Net-Net SBC receives a 401 or 407 message from the registrar in response to one of the following conditions, the endpoint attempting authentication is demoted.

- endpoint fails to authenticate with the Net-Net SBC using SIP HTTP digest authentication
- endpoint fails to authenticate with the Net-Net SBC using INVITE message when registration-caching is disabled

Configuring Endpoint Demotion on CAC Failures

To configure endpoint demotion on CAC failures:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm from the table and click **Edit**.
4. Click the Access control tab at the top of the content area.

5. Select an entry in the Access control list and click **Edit**
6. **CAC failure threshold**—Enter the number of CAC failures for any single endpoint that will demote it from the trusted queue to the untrusted queue. The default value is 0.

CAC failure threshold

7. **Untrusted CAC failure threshold**—Enter the number of CAC failures for any single endpoint that will demote it from the untrusted queue to the denied queue. The default value is 0.

Untrusted CAC failure threshold

8. Click **Apply**.

SRTP-SDES Configuration

The Secure Real-Time Transport Protocol (SRTP) provides encryption and authentication to the call content and call signalling streams. Authentication provides assurance that packets are from the purported source, and that they (the packets) have not been tampered with during transmission. Encryption provides assurance that the call content and associated signalling has remained private during transmission.

RTP and RTCP traffic are encrypted as described in RFC3711, *The Secure Real-time Transport Protocol (SRTP)*. The negotiation and establishment of keys and other cryptographic materials that support SRTP is described in RFC4568, *Session Description Protocol (SDP) Security Description for Media Streams*. Cryptographic parameters are established with only a single message or in single round-trip exchange using the offer/answer model defined in RFC 3264, *An Offer/Answer Model with the Session Description Protocol*.

Net-Net SBCs provide support for an initial SRTP/SDES version that implements encryption of the call content and signalling streams. Authentication of media streams will be added to a subsequent release.

The following SDP Offer and Answer illustrate the RFC4568 exchange.

Offer:

```
v=0
o=sam 2890844526 2890842807 IN IP4 10.47.16.5
s=SRTP Discussion
i=A discussion of Secure RTP
u=http://www.example.com/seminars/srtp.pdf
e=marge@example.com (Marge Simpson)
c=IN IP4 168.2.17.12
t=2873397496 2873404696
m=audio 49170 RTP/SAVP 0
a=crypto:1 AES_CM_128_HMAC_SHA1_80
inline:WVNfX19zZW1jdGwgKCkgewkyMjA7fQp9CnVubGVz|2^20|1:4
```

Answer:

```
v=0
o=jill 25690844 8070842634 IN IP4 10.47.16.5
s=SRTP Discussion
i=A discussion of Secure RTP
```

```

u=http://www.example.com/seminars/srtp.pdf
e=homer@example.com (Homer Simpson)
c=IN IP4 168.2.17.11
t=2873397526 2873405696
m=audio 32640 RTP/SAVP 0
a=crypto:1 AES_CM_128_HMAC_SHA1_80
inline:PS1uQCVeeCFCaVmcjKpPywJNWhcYD0mXXtaVBR|2^20|1:4

```

A media-level SDP attribute, **crypto**, describes the cryptographic suite, key parameters, and session parameters for the preceding unicast media line. The **crypto** attributes only appear at the SDP media level (not at the session level). The format of the crypto attribute is as follows:

```
a=crypto: tag crypto-suite key-parameter [session-parameters]
```

- **tag** is a decimal number that identifies a particular crypto attribute.
- **crypto-suite** describes that encryption and authentication algorithms, either AES_CM_128_HMAC_SHA1_80 or AES_CM_128_HMAC_SHA1_32.

- **key-parameter** provides one or more sets of keying material for the selected crypto-suite and it has following format.

```
"inline:" <key||salt> ["|" lifetime] ["|" MKI ":" length]
```

- **key||salt** provides a base64-encoded concatenated master key and salt.
- **lifetime** provides the master key lifetime (maximum number of SRTP or SRTCP packets encrypted using this master key).
- **MKI:length** provides the Master Key Index (MKI) value and the MKI length.
- **session-parameters** defines a set of optional parameters that may be used to override SRTP session defaults for the SRTP and SRTCP streams.

UNENCRYPTED_SRTP — SRTP messages are not encrypted

UNENCRYPTED_SRTCP — SRTCP messages are not encrypted

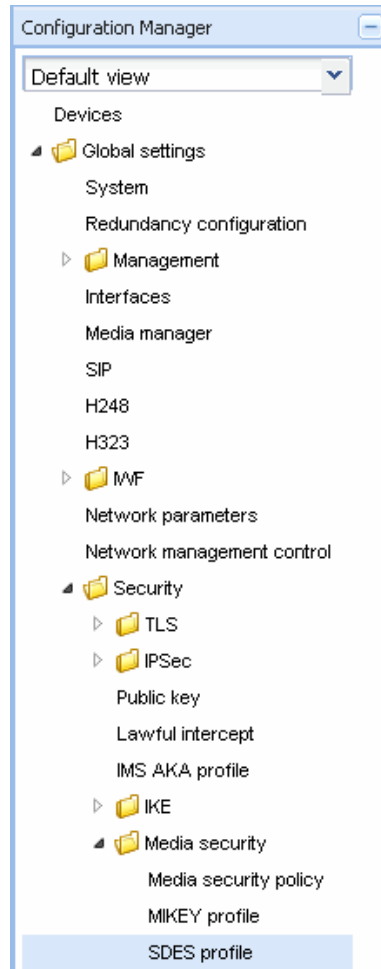
UNAUTHENTICATED_SRTP — SRTP messages are not authenticated

From the example above, the offerer supports AES crypto suite with MKI of 1. The answer accepts the offer by replying the SDP with its own crypto attribute. In this case, the session would use the AES_CM_128_HMAC_SHA1_80 crypto suite for the RTP and RTCP traffic.

Configuring an SDES Profile

To configure an SDES profile:

1. Expand the Configuration Manager slider > Global settings > Security > Media security.



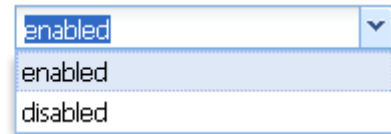
2. Click SDES profile. The SDES profile table appears in the content manager.
3. In the SDES profile table, click **Add**.
4. **Name**—Enter a unique name for this SDES profile.
5. Click **Apply**.
6. Select the SDES profile from the table and click **Edit**.
7. **SRTP authentication**—Select enabled from the combination box to enable the encryption of RTP packets. The valid values are:
 - **enabled** (default): The Net-Net SBC offers RTP encryption, and rejects an answer that contains an UNENCRYPTED_SRTP session parameter in the crypto attribute
 - **disabled**: The Net-Net SBC does not offer RTP encryption and includes an UNENCRYPTED_SRTP session parameter in the SDP crypto attribute; it accepts an answer that contains an UNENCRYPTED_SRTP session parameter.
8. **SRTP encryption**—Select enabled from the combination box to enable the encryption of RTP packets. The valid values are:

- **enabled** (default): The Net-Net SBC offers RTP encryption, and rejects an answer that contains an UNENCRYPTED_SRTP session parameter in the crypto attribute
 - **disabled**: The Net-Net SBC does not offer RTP encryption and includes an UNENCRYPTED_SRTP session parameter in the SDP crypto attribute; it accepts an answer that contains an UNENCRYPTED_SRTP session parameter.
9. **SRTCP encryption**—Select enabled from the combination box to enable the encryption of RTCP packets. The valid values are:
- **enabled** (default): The Net-Net SBC offers RTCP encryption, and rejects an answer that contains an UNENCRYPTED_SRTCP session parameter in the crypto attribute.
 - **disabled**: The Net-Net SBC does not offer RTCP encryption and includes an UNENCRYPTED_SRTCP session parameter in the SDP crypto attribute; it accepts an answer that contains an UNENCRYPTED_SRTCP session parameter.
10. **Master key identifier**—Select enabled from the combination box to enable the inclusion of the MKI:length field in the SDP crypto attribute.

The master key identifier (MKI) is an optional field within the SDP crypto attribute that uniquely identifies the SRTP stream within an SRTP session. MKI is expressed as a pair of decimal numbers in the form `|mki:mki_length|` where `mki` is the MKI value and `mki_length` its length in bytes. The valid values are:

- **enabled**: A four-byte MKI field is sent within the crypto attribute
- **disabled** (default): No MKI field is sent

Master key identifier



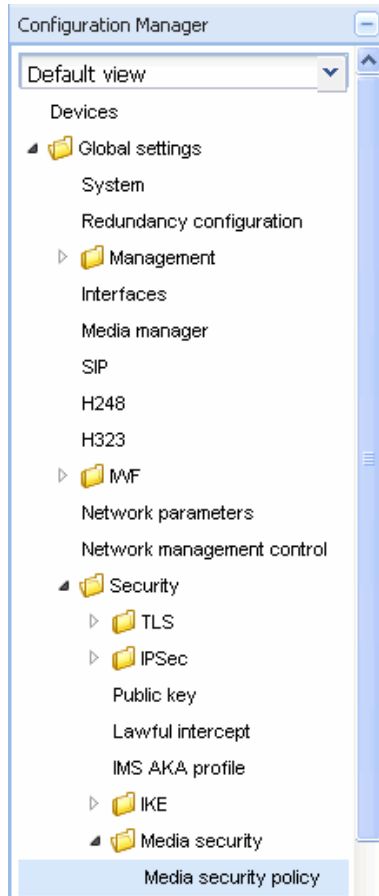
A screenshot of a web-based configuration interface. It shows a label 'Master key identifier' followed by a dropdown menu. The dropdown menu is open, showing two options: 'enabled' and 'disabled'. The 'enabled' option is currently selected and highlighted in blue.

11. Click **Apply**. The SDES profile is added to the SDES profile table.
12. Repeat steps 4 through 12 to configure additional SDES profiles.
13. Click **Apply**.

Configuring Media Security Policy

To configure a media security policy:

1. Expand the Configuration Manager slider > Global settings > Security > Media security.



2. Click Media security policy. The Media security policy table appears in the content area.
3. Click **Add**. The Add Media security policy dialog box appears.
4. **Name**—Enter a unique name for this media security policy profile.

Add Media security policy

*Name (Unique required)

5. Select the policy from the table and click **Edit**.
6. **Pass through** (*Optional*)—Select enabled or disabled from the combination box.
 - **enabled**: The User Agent (UA) endpoints negotiate security parameters between each other; consequently, the Net-Net SBC simply passes SRTP traffic between the two endpoints
 - **disabled** (default): The Net-Net SBC disallows end-to-end negotiation and initiates and terminates SRTP tunnels with both endpoints

Pass through

▼

enabled

disabled

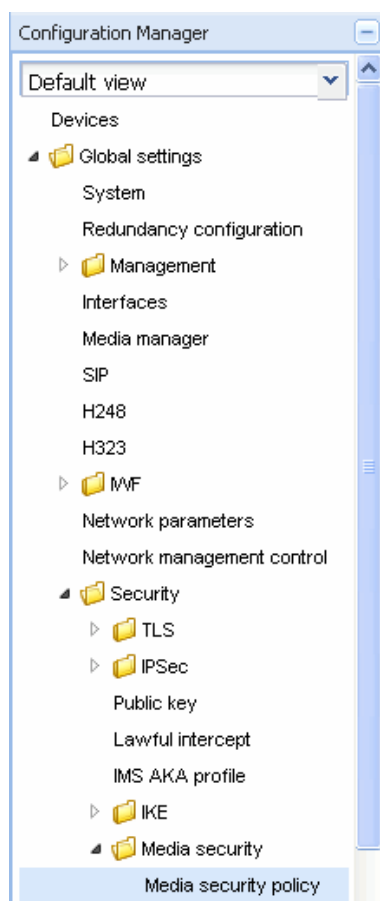
Configuring Inbound Parameters

- Click **Apply**.

You specify security parameters applied to the inbound (toward the access network) call leg.

To configure inbound parameters for a media security policy:

- Expand the Configuration Manager slider > Global settings > Security > Media security.



- Click Media security policy. The Media security policy table appears in the content area.
- Click the media security policy you want to configure in the Media security policy table and click **Edit**.
- Click **Add** beneath the Inbound table. The Add inbound dialog box appears.
- Protocol**—Click **sdes** as the key encryption protocol for SDES encryption in the combination box.
- Profile**—Enter the name of the SDES profile associated with the inbound leg of this media security policy.
- Mode**—Click the real time transport protocol in the combination box.

The **mode** specifies what transport protocol (RTP or SRTP) is allowed/used in the media descriptions of an offer SDP when this media-security-policy is in effect.

For the incoming direction, if a media description has no transport protocol that matches the value of this field, the INVITE message will be rejected with the 488 Not Acceptable Here response.



The 'Add inbound' dialog box contains the following fields:

- Protocol:** A dropdown menu with 'mikey' selected and '(Default:none)' as a hint.
- Mode:** A dropdown menu with 'srtp' selected and '(Default:rtp)' as a hint.
- Profile:** A text field containing 'mediaSecPolicy'.
- Configuration Guidelines:** A button with a dropdown arrow.
- Buttons:** 'Apply' and 'Cancel' buttons at the bottom.

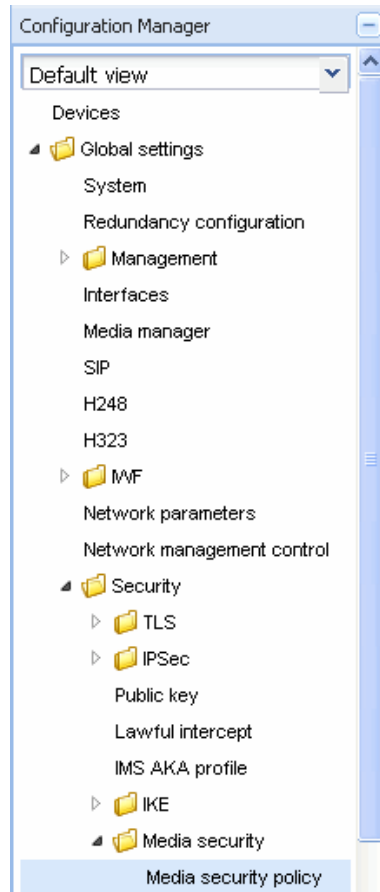
8. Click **Apply**.
9. Click **Apply**.

Configuring Outbound Parameters

You specify security parameters applied to the outbound (toward the core network) call leg.

To configure outbound parameters for a media security policy:

1. Expand the Configuration Manager slider > Global settings > Security > Media security.



2. Click Media security policy. The Media security policy table appears in the content area.

3. Click the media security policy you want to configure in the Media security policy table and click **Edit**.
4. Click **Add** beneath the Outbound table. The Add outbound dialog box appears.
5. **Protocol**—Click **sdes** as the key encryption protocol for SDES encryption in the combination box. The valid values are **mikey**, **sdes**, and **none**.
6. **Profile**—Enter the name of the SDES profile associated with the outbound leg of this media security policy.
7. **Mode**—Click the real time transport protocol in the combination box.

The **mode** specifies what transport protocol (RTP or SRTP) is allowed/used in the media descriptions of an offer SDP when this media security policy is in effect.

For the outgoing direction, the Net-Net SBC will send an offer SDP with media descriptions that have the transport protocol specified in this field.

8. Click **Apply**.
9. Click **Apply**.

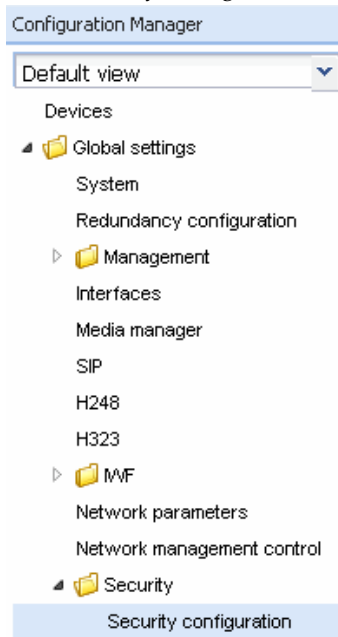
FIPS 140-2 Level 2

Configuring FIPS 140-2 Level2

To configure FIPS 140-2 Level 2:

1. Expand the Configuration Manager slider > Global settings > Security.

- Click Security configuration.



- Image integrity value**—Enter the Acme Packet generated HMAC-SHA256 string value for this field.

Image integrity value

- Click **Apply**.

SSH Public Key Support

SSHv2 Public Key Authentication

Net-Net Central supports viewing public keys. In order to import, generate or delete public keys, you must use the ACLI.

The Net-Net SBC supports importing, generating, viewing, and deleting public keys used for authentication of SSHv2 sessions from administrative remote users.

Note: You must be in superuser mode to perform this process in the ACLI.

Importing an SSH Host Key

Importing a host key requires access to the SFTP server or servers. Access is generally most easily accomplished with a terminal emulation program such as PuTTY, SecureCRT, or TeraTerm.

ACLI Instructions and Examples

You import an SSHv2 host key from the ACLI.

To import an SSHv2 host key:

- Access the SSH file system on a configured SFTP server using a terminal emulation program.
- Copy the server's base64 encoded public file, including the Begin and End markers as specified by RFC 4716, *The Secure Shell (SSH) Public Key File Format*.

For OpenSSH implementations, host files are generally found at */etc/ssh/ssh_host_dsa_key.pub*, or *etc/ssh/ssh_host_rsa.pub*. Other SSH implementations can differ.

3. Use the **security ssh-pub-key import known-host <name>** command (where **<name>** is an alias or handle assigned to the imported host key, generally the server name or a description of the server function) to import the host key to the SBC.
4. Paste the public key with the bracketing Begin and End markers at the cursor point.
5. Enter a semi-colon (;) to signal the end of the imported host key.
6. Save and activate the configuration.

The entire import sequence is shown below.

```
ACMEPACKET# security ssh-pub-key import known-host fedallah
IMPORTANT:
    Please paste ssh public key in the format defined in rfc4716.
    Terminate the key with ";" to exit.....

---- BEGIN SSH2 PUBLIC KEY ----
Comment: "2048-bit RSA, converted from OpenSSH by klee@acme54"
AAAAB3NzaC1yc2EAAAABIwAAAQEA70Bf08jJe7MSMgerjDTgZpbPblrX4n17LQJgPC7cLL
cDGEtKSivt5MjcSav3v6AEN2pYZih0xd2Zzismpoo019kkJ56s/IjGstEzqXMKHKur9mBV
qvqIEOTqbowEi5sz2AP31GUjQTCKZRF1X0Qx8A44vHZCum93/jfNRsnWQ1mhHmaZMmT2LS
h0r4J/Nlp+vpsvpdrolV6Ftz5eiVfgocxrDrjNcVtsAMYLBpDdL6e9XebQzGSS92TPuKP/
yqzLJ2G5NVFhxdw5i+FvdHz1vBdvB505y2QPj/iz1u3TA/307tyntB0b7beDyIrg64AzC8
G7E3AGiH49LnBtlQf/aw==
---- END SSH2 PUBLIC KEY ----
;
SSH public key imported successfully....
WARNING: Configuration changed, run "save-config" command to save it
and run "activate-config" to activate the changes
ACMEPACKET# save config
checking configuration
-----
...
...
...
-----
Save-Config received, processing.
waiting for request to finish
Request to 'SAVE-CONFIG' has Finished,
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot activate'.
ACMEPACKET# activate config
Activate-Config received, processing.
waiting for request to finish
SD is not QOS-capable
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
ACMEPACKET#
```

Importing an SSH Public Key

Prior to using SSH public key-based authentication you must import a copy of the public key of each user who will authenticate using this method. The public key identifies the user as a trusted entity when the Acme Packet Net-Net SBC performs authentication.

During the SSH login, the user presents its public key to the SBC. Upon receiving the offered public key, the SBC validates it against the previously obtained trusted copy of the key to identify and authenticate the user.

Importing a public key requires access to the device on which the public key was generated, or on which it is currently stored with its associated private key. Access is generally attained with a terminal emulation program such as PuTTY, SecureCRT, or TeraTerm.

Importing a Public Key from the ACLI

Note: You import a public key from the ACLI.

To import a public key:

1. Access the system from which the public key will be obtained using a terminal emulation program.
2. Copy the base64 encoded public key, including the Begin and End markers as specified by RFC 4716, *The Secure Shell (SSH) Public Key File Format*.
3. Use the **security ssh-pub-key import authorized-key <name>** command (where <name> is an alias or handle assigned to the imported public key, often the user's name) to import the public key to the SBC. For example:

```
ACMEPACKET# security ssh-pub-key import authorized-key key1
```

Use the optional "<admin>" argument to import a public key who will be authorized for admin privileges. For example:

```
ACMEPACKET# security ssh-pub-key import authorized-key key1 admin
```

4. Paste the public key with the bracketing Begin and End markers at the cursor point.
5. Enter a semi-colon (;) to signal the end of the imported host key.
6. Save and activate the configuration.

The entire import sequence is shown below.

```
ACMEPACKET# security ssh-pub-key import authorized-key Matilda
admin
```

IMPORTANT:

Please paste ssh public key in the format defined in rfc4716.
Terminate the key with ";" to exit.....

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "1024-bit RSA, converted from OpenSSH by abhat@acme74"
AAAAB3NzaC1yc2EAAAABIwAAAIEAxcYTV595VqdHy12P+mIZBlpe0Zx9sX/mSAFihDJYdL
qJIWdiZuSmny8HZIXTIC6na62iD25mLEdyLhly0uknkYBCU7UsLwmx4dLDyHTbrQH3b1q
3Tb8auz97/J1p4pw39PT42CoR0DzPBrXJV+OglNE/83C1y0SSJ8Bjc9LEwE=
---- END SSH2 PUBLIC KEY ----:
SSH public key imported successfully...
WARNING: Configuration changed, run "save-config" command to save it
and run "activate-config" to activate the changes
ACMEPACKET# save config
checking configuration
-----
...
...
...
-----
```



```

Save-Config received, processing.
waiting for request to finish
Request to 'SAVE-CONFIG' has Finished,
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot activate'.
ACMEPACKET# activate config
Activate-Config received, processing.
waiting for request to finish
SD is not QOS-capable
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
ACMEPACKET#

```

Generating an SSH Key Pair

To generate an SSH key pair you must initially configure a public key record which will serve as a container for the generated key pair.

Configuring a Public Record from the ACLI

Note: You configure a public key record from the ACLI.

To configure a public key record:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)#
```
2. Type **security** and press <Enter>.


```
ACMEPACKET(configure)# security
ACMEPACKET(security)#
```
3. Type **public-key** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(security)# public-key
ACMEPACKET(public-key)#
```
4. **name**—Enter the name of this public key.


```
ACMEPACKET(public-key)# name tashtego
```
5. **type**—Select the type of key you want to create. Valid values are **rsa** and **dsa**. The default value is **rsa**.
6. **size**—Enter the size of the key you are creating. Valid key sizes are **512**, **1024**, or **2048**. The default value is **1024**.
7. Save and activate the configuration.

You generate an SSH key pair from the ACLI.

To generate an SSH key pair:

1. Enter the **security ssh-pub-key generate** command along with the name of the public key you created in the previous step as follows.


```
ACMEPACKET# security ssh-pub-key generate tashtego
Please wait...
public-key 'tashtego' (RFC 4716/SECSH format):
---- BEGIN SSH2 PUBLIC KEY ----
```

```
Comment: "1024-bit rsa"
```

```
AAAAB3NzaC1yc2EAAAABIwAAAIEArZEP1/WiYsdGd/Pi8V6pnSwV4cVG4U+jV
OwiSwNJCC9Nk82/FKYleLZevy9D3lrZ8ytvu+sCYy0fNk4nwwz20c2N+r86kD
ru88JkUqpelJDx1AR718Icpr7ZaAx2L+e7cpyRSXCgbQR7rXu2H3bp9Jc0VhR
2fmkcLmrGAir7Gnc=
```

```
----- END SSH2 PUBLIC KEY -----
```

SSH public-key pair generated successfully....

WARNING: Configuration changed, run "save config" command to save it and run "activate config" to activate the changes

ACMEPACKET#

2. Copy the base64-encoded public key. Copy the public key including the bracketing Begin and End markers. Shortly you will paste the public key to one or more SFTP servers.
3. Save and activate the configuration.

Copying a Client Key or an SSH or SFTP Server

Copying the client public key to the SFTP server requires server access generally using a terminal emulation program such as PuTTY, SecureCRT, or TeraTerm.

1. Use a terminal emulation program to access the SSH file system on a configured SFTP server.
2. Copy the client key to the SFTP server.

On OpenSSH implementations, public keys are usually stored in the `~/.ssh/authorized_keys` file. Each line in this file (1) is empty, (2) starts with a pound(#) character (indicating a comment), or (3) contains a single public key.

Refer to the `sshd` manual pages for additional information regarding file format.

Use a text editor such as *vi* or *emacs* to open the file and paste the public key to the tail of the `authorized_keys` file.

For SSH implementations other than OpenSSH, consult the system administrator for file structure details.

Viewing SSH Imported Keys

You can use the **show security ssh-pub-key** command to display information about SSH keys imported to the SBC with the **ssh-pub-key** command; you cannot display information about keys generated by the **ssh-pub-key** command. You can view either a brief version or detailed information about the certificates.

Viewing a Brief SSH Imported Key Version

You view the brief SSH imported key version from the ACLI.

When you use the **show security ssh-pub-key brief [name]** command, the following information is displayed:

- **login-name**—Contains the name assigned to the RSA or DSA public key when it was first imported
- **finger-print**—Contains the output of an MD5 hash computed across the base64-encoded public key
- **finger-print-raw**—Contains the output of an MD5 hash computed across the binary form of the public key

To view the brief version for all imported SSH public keys:

```

ACMEPACKET# show security ssh-pub-key brief
login-name:
  acme74
finger-print:
  51:2f:f1:dd:79:9e:64:85:6f:22:3d:fe:99:1f:c8:21
finger-print-raw:
  0a:ba:d8:ef:bb:b4:41:d0:dd:42:b0:6f:6b:50:97:31

login-name:
  fedallah
finger-print:
  c4:a0:eb:79:5b:19:01:f1:9c:50:b3:6a:6a:7c:63:d5
finger-print-raw:
  ac:27:58:14:a9:7e:83:fd:61:c0:5c:c8:ef:78:e0:9c
ACMEPACKET#

```

To view the brief version for a specified imported SSH public key:

```

ACMEPACKET# show security ssh-pub-key brief fedallah
login-name:
  fedallah
finger-print:
  c4:a0:eb:79:5b:19:01:f1:9c:50:b3:6a:6a:7c:63:d5
finger-print-raw:
  ac:27:58:14:a9:7e:83:fd:61:c0:5c:c8:ef:78:e0:9c
ACMEPACKET#

```

Viewing a Detailed SSH Imported Key Version

You view a detailed SSH imported key version from the ACLI.

When you use the **show security ssh-pub-key detail <name>** command, the following information is displayed:

For an RSA key:

- **host-name**—Contains the name assigned to the RSA key when it was first imported
- **finger-print**—Contains the output of an MD5 hash computed across the base64-encoded RSA public key
- **finger-print-raw**—Contains the output of an MD5 hash computed across the binary form of the RSA public key
- **public key**—Contains the base64-encoded RSA key
- **modulus**—Contains the hexadecimal modulus (256) of the RSA key
- **exponent**—Contains an integer value that is used during the RSA key generation algorithm. Commonly used values are 17 and 65537. A prime exponent greater than 2 is generally used for more efficient key generation.

For a DSA key:

- host name—Contains the name assigned to the DSA public key when it was first imported
- comment—Contains any comments associated with the DSA key
- finger-print—Contains the output of an MD5 hash computed across the base64-encoded DSA public key
- finger-print-raw—Contains the output of an MD5 hash computed across the binary form of the DSA public key
- public key—Contains the base64 encoded DSA key
- p—Contains the first of two prime numbers used for key generation
- q—Contains the second of two prime numbers used for key generation
- g—Contains an integer that together with p and q are the inputs to the DSA key generation algorithm

To view the detailed version for a specific RSA key:

ACMEPACKET# **show security ssh-pub-key detail fedallah**

host-name:

fedallah

comment:

"2048-bit RSA, converted from OpenSSH by klee@acme54"

finger-print:

c4:a0:eb:79:5b:19:01:f1:9c:50:b3:6a:6a:7c:63:d5

finger-print-raw:

ac:27:58:14:a9:7e:83:fd:61:c0:5c:c8:ef:78:e0:9c

pub-key:

AAAAB3NzaC1yc2EAAAABIwAAQEA70Bf08jJe7MSMgerjDTgZpbPblrX4n17LQJgP
C7cLLcDGEtKSivt5MjcSav3v6AEN2pYZih0xd2Zzispoo019kkJ56s/IjGstEzqX
MKHKUr9mBVqvqIE0TqbowEi5sz2AP31GUjQTCKZRF1X0Qx8A44vHZCum93/jfNRsn
WQ1mhHmaZMMt2LSh0r4J/Nlp+vpsvpdroLV6Ftz5eiVfgocxrDrjNcVtsAMyLBpDd
L6e9XebQzGSS92TPuKP/yqzLJ2G5NVFhxdw5i+FvdHz1vBdvB505y2QPj/iz1u3TA
/307tyntB0b7beDyIrg64AzC8G7E3AGiH49LnBt lQf/aw==

modulus: (256)

ECE05FD3C8C97BB3123207AB8C34E06696CF6E5AD7E27D7B2D02603C2EDC94B70
3184B4A4A256DE4C8DC49ABF7BFA004376A5866284EC5DD99CE2B26A68A34D7D9
24279EACFC88C6B2D133A9730A1CA52BF66055AAFA8810E4EA6E8C048B9B33D80
3F7D4652341308A6511755CE431F00E38BC7642BA6F77FE37CD46C9D64359A11E
66993264F62D284EAF827F365A7EBE9B2FA5DAE8955E85B73E5E8957E0A1CC6B0
EB8CD715B6C00CC8B0690DD2FA7BD5DE6D0CC6492F764CFB8A3FFCAACCB2761B9
355161C5DC398BE16F747CF5BC176F079D39CB640F8FF8B3D6EDD303FDCEEDCA
7B4139BEDB783C88AE0EB803373C1BB137006887E3D2E706D9507FF6B

exponent: (1)

23

ACMEPACKET#

To view the detailed version for a specific DSA key:

ACMEPACKET# **show security ssh-pub-key detail acme74**

host-name:

acme74

comment:

DSA Public Key

finger-print:

51:2f:f1:dd:79:9e:64:85:6f:22:3d:fe:99:1f:c8:21

finger-print-raw:

0a:ba:d8:ef:bb:b4:41:d0:dd:42:b0:6f:6b:50:97:31

pub-key:

```
AAAAB3NzaC1kc3MAAACBAPY8Z0HY2yFSJA6XYC9HRwNHxaehvx5w0J0rzZdzoS0Xx
bETW6ToHv8D1UJ/z+zHo9Fiko5XybZnDIaBDHtblQ+Yp7StxyltHnXF1YLFKD1G4T
6JYrdHYI140m1eg9e4NnCRleaQZPF3UGfZia6bXrGTQf3gJq2e7Yisk/gF+1VAAA
AFQDb8D5cvwHWTZDPfX0D2s9Rd7NBvQAAAIEAlN92+Bb7D4KLYk3IwRbXblwXdkPg
gA4pfdtW9vGfJ0/RHd+NjB4eo1D+0dix6tXwYGN7PKS5R/FXPNwxHPapcj9uL1Jn2
AWQ2dsknf+i/FAAvioUPkmdMc0zuWoS0EsSNhVDtX3WdvVcGcBq9cetzrtOKW0ocJ
mJ80qadxTRHtUAAACBAN7CY+KKv1gHpRzFwdQm7HK9bb1LAo2KwaoXnadFgeptNBQ
eSXG1v0+JsvphVMBJc9HSn24VYtYtsMu74qXviYjziVucWKjjKEb11juqnF0GDlB3
VVMxHLMxnAz643WK42Z7dLM5sY29oueZv4Xz2PuMch5VGPP+CDqzCM4l0WgV
```

p: (128)

```
F63C64E1D8DB2152240E97602F47470347C5A7A1BF1E70389D2BCD9773A12397C
5B1135BA4E81EFF03D5427FCFECC7A3D162928E57C9B6670C86810C7B5B950F98
A7B4ADC7296D1E75C5D582DF283D46E13E8962B747608D783A6D5E83D7B836709
195E6AAA193C5DD419F6626BA6D7AC64D07F7809AB67BB622B24FE017ED55
```

q: (20)

```
DBF03E5CBF01D64D90CF7D7D03DACF5177B341BD
```

g: (128)

```
94DF76F816FB0F828B624DC8C116D76E5C177643E0800E297DDB56F6F19F274FD
11DDF8D8C1E1EA350FED1D8B1EAD5F060637B3CA4B947F1573CDC311CF6A9723F
6E2F5267D80590D9DB249DFFA2FC5000BE2A143E499D31CD33B96A12384B12361
543B57DD676F55C19C06AF5C7ADCEBB4E2963A8709989F34A9A7714D11ED5
```

pub_key: (128)

```
DEC263E28ABF5807A51CC5C1D426EC72BD6DBD4B028D8AC1AA179DA74581EA6D3
4141E4971B5BCEF89B2FA6154C04973D1D29F6E1562D62DB0CBBBE2A5EF8988F3
895B9C58A8E32846F5D63BAA9C5D060E50775559B11CB9B19C0CFAE3758AE3667
B74B339B18DBDA2E7B3BF85F3D8FB8C721E5518F3FE083AB308CE25A16815
```

ACMEPACKET#

To view the detailed version for all SSH imported keys:

ACMEPACKET# **show security ssh-pub-key detail**

...

...

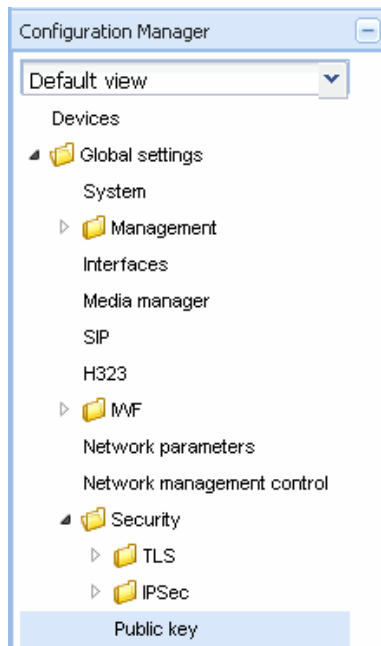
...

ACMEPACKET#

**Viewing a Public Key
Using Net-Net Central**

To view public keys using Net-Net Central:

1. Expand the Configuration Manager slider > Global settings > Security.



2. Click Public key.
3. Select the public you would like to view from the table.
4. Click **View**.

Deleting a Public Key Record

This section shows you how to delete a public key record. Note that this process requires you to save and activate your configuration.

Deleting a Public Key using the ACLI

Note: You delete an SSHv2 public key record in the ACLI.

To delete an SSHv2 public key record:

1. In Superuser mode, type the command **security ssh-public-key delete**, then a <Space> and the login-name (found in both brief and detail **show security public-key** commands) corresponding to the public key you want to delete.

The Net-Net SBC confirms you have successfully deleted the key, and then reminds you to save your configuration.

After you complete this procedure, you can confirm the public key has been deleted by using either of the show security **ssh-pub-key** commands.

```
ACMEPACKET# security ssh-pub-key delete jdoe
SSH public key deleted successfully...
WARNING: Configuration changed, run "save-config" command.
```

```
ACMEPACKET# security ssh-pub-key delete jdoe
record (jdoe) does not exist
```

2. Save and activate your configuration.

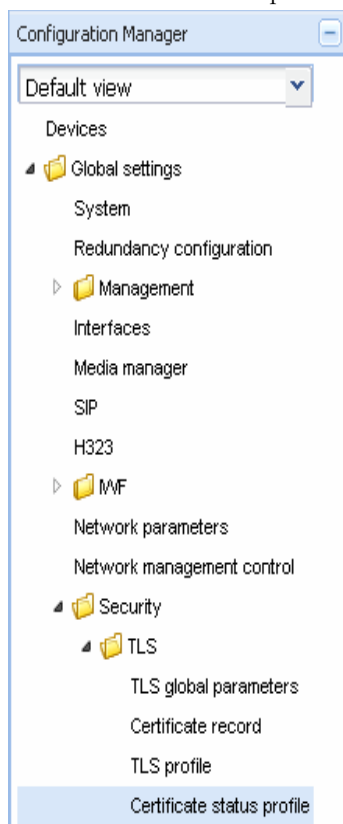
TLS IPv6

Configuring TLS for IPv6

Note: You must configure this feature in the ACLI. You can only view this parameter using Net-Net Central.

To view TLS configuration:

3. Expand the Configuration Manager slider -> Global settings -> Security -> TLS.
4. Click Certificate status profile.



From here, you can view the Name, IP address, and port for certificate status profiles.

Unreachable OCSR

2265

With OCSP enabled, the client implementation running on the Session Director (SD) supports message exchange between the SD and an OCSR as specified in RFC 2560, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*. The SD contacts the OCSR whenever a remote client attempts to establish an SSL/TLS connection with the SD. The SD sends a request to the OCSR to check the current status of the certificate presented by the remote client. The SD suspends processing of the SSL/TLS connection request pending receipt of the OCSR response. In previous releases (prior to Version 7.3), a *good* OCSR response resulted in the establishment of a secure SSL/TLS connection. A *revoked* or *unknown* OCSR response, or the failure to reach an OCSR, resulted in the rejection of the connection attempt.

This behavior, which adheres to the requirements of RFC 2560, conflicts with the requirements of Section 5.4.6.2.1.6.4.a.i of UCR 2008 which requires an OCSP client to attempt authentication of remote clients in the event of an unreachable OCSR.

There is no configuration for this feature. It is enabled automatically.

Overriding the Default Behavior using the ACLI

A variable is added under each TLS profile object that allows you the option of overriding the default behavior of the OCSP feature. When enabled, the variable will cause SSL/TLS connections to be established if the SD decides the client certificate is valid, but the OCSR is unreachable.

Note: You cannot override the default behavior using Net-Net Central. The **Ignore deade responder** parameter is view-only and must be enabled or disabled using the ACLI.

Online Certificate Status Protocol

The Online Certificate Status Protocol (OCSP) is defined in RFC 2560, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*. The protocol enables users to determine the revocation state of a specific certificate, and may provide a more efficient source of revocation information than is possible with Certificate Revocation Lists (CRL).

The protocol specifies the data exchanged between an OCSP client (for example, the Net-Net SBC) and an OCSP responder, the Certification Authority (CA), or its delegate, that issued the target certificate. An OCSP client issues a request to an OCSP responder and suspends acceptance of the certificate in question until the responder replies with a certificate status.

Certificate status is reported as

- good
- revoked
- unknown

good indicates a positive response to the status inquiry. At a minimum, this positive response indicates that the certificate is not revoked, but does not necessarily mean that the certificate was ever issued or that the time at which the response was produced is within the certificate's validity interval.

revoked indicates that the certificate has been revoked, either permanently or temporarily.

unknown indicates that the responder cannot identify the certificate.

Caveats

OCSP is currently supported only on TLS interfaces; it is not currently supported for use with IKEv1 and IKEv2.

OCSP configuration consists of:

1. Configuring one or more certificate status profiles; each profile contains information needed to contact a specific OCSP responder.
2. Enabling certificate revocation checking by assigning a certificate status profile to a previously configured TLS profile.

Creating a Certificate Status Profile using the ACLI

Note: This feature is configured in the ACLI only and is read-only in Net-Net Central.

To create a certificate status profile using the ACLI:

1. From superuser mode, use the following command sequence to access *cert-status-profile* configuration mode. While in this mode, you provide the information required to access one or more OCSP responders.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# security
ACMEPACKET(security)# cert-status-profile
ACMEPACKET(cert-status-profile)#
```
2. Use the required **name** parameter to identify this *cert-status-profile* instance — each profile instance provides configuration data for a specific OCSP responder. **name** is used to distinguish between multiple profile instances.
3. Use the required **ip-address** parameter to specify the IPv4 address of the OCSP responder.
4. Use the optional **port** parameter to specify the destination port.

In the absence of an explicitly configured value, the default port number of 80 is used.
5. Use the optional **realm-id** parameter to specify the realm used to transmit OCSP requests.

In the absence of an explicitly configured value, the default specifies service across the *wancom0* interface.
6. Use the optional **requester-cert** parameter only if OCSP requests are signed; ignore this parameter if requests are not signed.

RFC 2560 does not require signed requests; however, local or CA policies can mandate digital signature..

7. Use the required **responder-cert** parameter to identify the certificate used to validate OCSP responses — a public key of the OCSP responder.

RFC 2560 requires that all OCSP responders digitally sign OCSP responses, and that OCSP clients validate incoming signatures.

Provide the name of the certificate configuration element that contains the certificate used to validate the signed OCSP response.

8. Use the optional **retry-count** parameter to specify the maximum number of times to retry an OCSP responder in the event of connection failure.

If the retry counter specified by this parameter is exceeded, the OCSP requester either contacts another responder (if multiple responders have been configured within this *cert-status-profile*) and quarantine the unavailable responder for a period defined the **dead-time** parameter.

In the absence of an explicitly configured value (an integer within the range 0 through 10), the default of 1 is used.

```
ACMEPACKET(cert-status-profile)# retry-count 2
ACMEPACKET(cert-status-profile)#
```

9. Use the optional **dead-time** parameter to specify the quarantine period imposed on an unavailable OCSP responder.

In the absence of an explicitly configured value (an integer within the range 0 through 3600 seconds), the default value (0) is used.

Customer implementations utilizing a single OCSP responder are encouraged to retain the default value, or to specify a brief quarantine period to prevent lengthy service outages.

10. Retain default values for the **type** and **trans-protocol** parameter to specify OCSP over an HTTP transport protocol.
11. Use **done**, **exit**, and **verify-config** to complete configuration of this *cert-status-profile* instance.
12. Repeat Steps 1 through 11 to configure additional certificate status profiles.

To enable certificate status checking:

1. Move to *tls-profile* configuration mode.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# security
ACMEPACKET(security)# tls-profile
ACMEPACKET(tls-profile)#
```

2. Use the required **cert-status-check** parameter to enable OCSP in conjunction with an existing TLS profile.
3. Use the required **cert-status-profile-list** parameter to assign one or more *cert-status-profiles* to the current TLS profile.

Each assigned *cert-status-profile* provides the information needed to access a single OCSP responder.

Use quotation marks to assign multiple OCSP responders. The following sequence assigns three *cert-status-profiles*, *VerisignClass3Designate*, *Verisign-1*, and *Thawte-1* to the TLS-1 profile.

4. Use **done**, **exit**, and **verify-config** to complete configuration.

Sample Configuration:

```

ACMEPACKET# configure terminal
ACMEPACKET(configure)# security
ACMEPACKET(security)# cert-status-profile
ACMEPACKET(cert-status-profile)# name VerisignClass3Designate
ACMEPACKET(cert-status-profile)# ip-address 192.168.7.100
ACMEPACKET(cert-status-profile)# responder-cert
VerisignClass3ValidateOCSP
ACMEPACKET(cert-status-profile)# done
ACMEPACKET(cert-status-profile)# exit
...
...
ACMEPACKET# configure terminal
ACMEPACKET(configure)# security
ACMEPACKET(security)# tls-profile
ACMEPACKET(tls-profile)# select
<name>:
1. TLS-1
2. TLS-2
3. TLS-3

selection: 1
ACMEPACKET(tls-profile)# cert-status-check enabled
ACMEPACKET(cert-status-profile)# cert-status-profile-list
"VerisignClass3Designate Verisign-1 Thawte-1"
ACMEPACKET(cert-status-profile)# done
ACMEPACKET(cert-status-profile)# exit

```

High Capacity SRTP SDES

Hardware-specific configuration required.

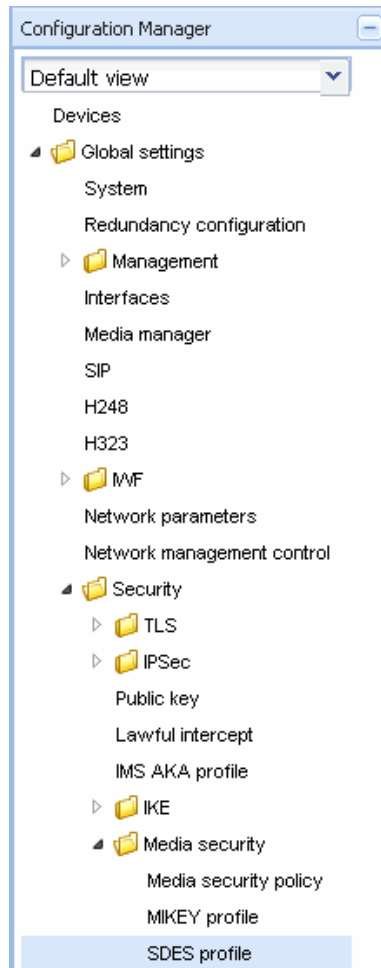
SRTP Support

The SDES profile now supports Aria Cipher values.

Configuring SRTP Support

To configure SRTP support:

1. Expand the Configuration Manager slider > Global settings > Security > Media security.
2. Click SDES profile.



3. Click the SDES profile you want to configure in the SDES profile table and click **Edit**.
4. **Crypto suites**—Click **Add**. The Add crypto suites dialog box appears.
5. Click one of the two Aria Cipher values in the drop down list. The Aria Cipher values are:
 - ARIA_CM_192_HMAC_SHA1_80
 - ARIA_CM_!(@_HMAC_SHA1_32
6. Click **Apply**. A Success message appears.
7. Click **OK**.

Intrusion Detection System

Release C6.2.1F2 provides an enhanced Intrusion Detection System referred to within this document as IDS Phase 2, which supplements the original IDS reporting and protection services first offered in the Net-Net S-C6.2.0 release. IDS Phase 2 provides enterprise users with additional tools to identify, monitor, and control suspicious, and possibly, malicious traffic patterns.

License Requirements

IDS Phase 2 requires the IDS Advanced Reporting license, which is different from the IDS license introduced in the Net-Net S-C6.2.0 release. The IDS Advanced Reporting license enables both the initial IDS services offered by S-C6.2.0 and the Phase 2 enhancements offered by C6.2.1F2. Access to new services described in this Technical Notice requires that the original IDS license be upgraded to the IDS Advanced Reporting license.

Rejected SIP Calls

IDS Phase 2 provides tools to monitor and record rejected SIP calls. A sudden or gradual increase in such calls can, but need not, indicate malicious intent.

IDS Phase 2 provides a global counter that increments with each SIP INVITE or REGISTER message that is rejected by the Acme Packet Net-Net SBC, and offers the option of generating a syslog message in response to call rejection.

Rejected Calls Counter

The rejected calls counter is a 32-bit global counter that records the total number of rejected SIP calls. For the initial (S-C6.2.1 release) such calls have been rejected by the SBC with the following response codes: 400, 403, 404, 405, 408, 413, 416, 417, 420, 423, 480, 481, 483, 484, 485, 488, 494, 500, 503, 505, and 604. These response codes may change with future software revisions.

The current value of the rejected calls counter is accessible via SNMP, Historical Data Recording (HDR), or the ACLI.

apSysMgmtGeneralObjects Table (1.3.6.1.4.1.9148.3.2.1.1)

Object Name	Object OID	Description
apSysSipTotalCallsRejected	1.3.6.1.4.1.9148.3.2.1.1.25	Global counter for SIP calls that are rejected by the SBC

The *sip-error* HDR collection group contains a new reporting field, *Call Rejects*, which contains the value of the global rejected calls counter.

Syslog Reporting of Rejected Calls

Users can choose to send a syslog message in response to the rejection of a SIP call. In the default state, rejected calls are not reported to syslog.

The Net-Net EMS label: **Allow syslog to be sent for call that gets rejected**, which is disabled by default, enables the generation of a syslog message in response to the rejection of a SIP call.

Syslog messages issued in response to call rejection contain the following call-related information.

- SIP status code indicating rejection cause
- SIP method name (INVITE or REGISTER)

- Reason for denial
- Realm of calling endpoint
- Applicable local response map
- Content of Reason header (if present)
- From URI of calling endpoint
- Target URI of called endpoint
- Source and Destination IP address and port
- Transport type

The following are sample syslog messages issued in response to call rejections.

```
Dec 8 06:05:42 172.30.70.119 deimos sipd[205bfee4] ERROR [IDS_LOG]INVITE from
source 172.16.18.100:5060 to dest 172.16.101.13:5060[UDP] realm=net172; From=sipp
<sip:sipp@172.16.18.100:5060>;tag=13890SIPpTag001;
target=sip:service@172.16.101.13:5060 rejected!; status=483 (Too Many Hops)
```

```
Dec 10 15:09:28 172.30.70.119 deimos sipd[2065ace8] ERROR [IDS_LOG]INVITE from
source 172.16.18.5:5060 to dest 172.16.101.13:5060[UDP] realm=net172; From=sipp
<sip:sipp@172.16.18.5:5060>;tag=10015SIPpTag001;
target=sip:service@172.16.101.13:5060 rejected!; status=488 (sdp-address-mismatch);
error=sdp address mismatch
```

IDS syslog messages that report rejected calls (introduced as part of IDS Phase 2 in S-C6.2.1) and those that report endpoint demotions (introduced as part of IDS Phase 1 in S-C6.2.0) now contain a string “IDS_LOG”, to facilitate their identification as IDS-related messages. With IDS Phase 2, IDS messages reporting either endpoint demotions or call rejections can be sent to specific, previously-configured syslog servers.

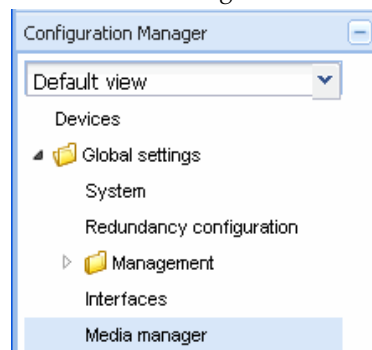
In topologies that include multiple syslog servers, use the following procedure to enable delivery of IDS-related messages to one or more specific syslog servers.

Enabling Syslog Reporting of Rejected Calls

Users can choose to send a syslog message in response to the rejection of a SIP call. In the default state, rejected calls are not reported to syslog.

To enable syslog reporting of rejected SIP calls:

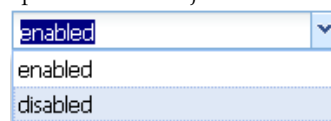
1. Expand the Configuration Manager slider -> Global settings.
2. Click Media manager.



3. Scroll to Syslog on call reject.

4. **Syslog on call reject**—Click **enabled** in the drop-down list to enable generation of a syslog message in response to the rejection of a SIP call.

Syslog on call reject

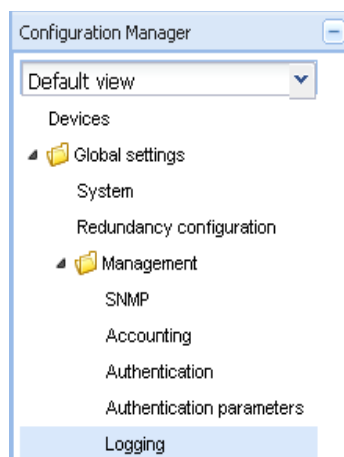


5. Click **Apply**.

Enabling Message Transfer to Specific Syslog Servers

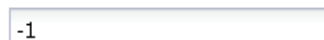
To enable message transfer to specific syslog servers:

1. Expand the Configuration Manager slider -> Global settings -> Management.
2. Click Logging.



3. Scroll to Syslog.
4. **IDS syslog facility**—Enter a value to enable message transfer to specific syslog servers. The default value, -1, disables selective message transfer. To enable transfer to a designated syslog server or servers, enter a value ranging from 0 through 23.

IDS syslog facility



5. Click **Apply**.

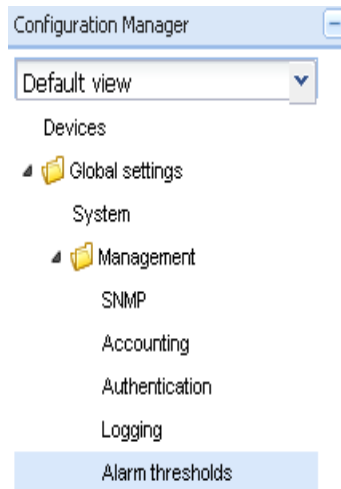
TCA Reporting of Denied Entries

You can configure a Threshold Crossing Alarm (TCA), which issues minor, major, and critical system alarms when the count of denied entries exceeds pre-configured values. For each issued alarm, the TCA also transmits an SNMP trap that reports the alarm state to remote SNMP agents.

Configuring Threshold Crossing Alarms

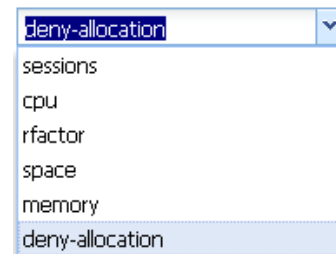
To configure a threshold crossing alarm:

1. Expand the Configuration Manager slider -> Global settings -> Management.
2. Click Alarm Thresholds..



3. Click **Add**. The Add alarm-threshold dialog box appears.
4. **Type**—Click **deny-allocation** in the drop-down list to specify the maximum number of denied entries allocated by the Net-Net SBC.

*Type



Threshold values that trigger alarm and SNMP trap generation are defined as percentages of the **deny-allocation** attribute.

5. Click **Apply**. A Success dialog box appears.
6. Click **OK**.
7. Click **Apply**.

After issuing a system alarm and accompanying SNMP trap, the TCA continues to monitor the number of denied entries. If the number of denied entries rises to the next threshold value, a new, and more severe, system alarm/SNMP trap is generated. If the number of denied entries falls below the current threshold level, and remains there for a period of at least 10 seconds, a new, and less severe system alarm/SNMP trap is generated.

Syslog Reporting of Denied Entries

Syslog reporting of endpoint demotions was introduced as part of IDS Phase 1 in S-C6.2.0. With IDS Phase 2 (S-C6.2.1 and later), such syslog messages contain the last SIP message from the endpoint that caused the transition to the denied state. If the included SIP message increases the length of the syslog beyond 1024 bytes, the SIP message is truncated so that the syslog is no larger than 1024 bytes.

CPU Load Limiting

The transmission of IDS-related system alarms and SNMP traps is disabled when the CPU utilization rate surpasses a configured threshold percentage. When the threshold is exceeded, a syslog message (MINOR level) announces the termination of IDS reporting. No additional syslog messages or SNMP traps are generated until the CPU utilization rate falls below the configured threshold. The resumption of IDS reporting is announced by another syslog message, also issued at the MINOR level.

By default the CPU utilization rate is 90%.

Denied Endpoints

IDS Phase 2 provides a denied endpoint counter that includes SIP and MGCP (Media Gateway Control Protocol) endpoints. The global counter value is available via SNMP or HDR.

The global counter value is available to SNMP under APSYSMGMT-MIB -> acmepacketMgmt -> apSystemManagementModule -> apSysMgmtMIBObjects -> apSysMgmtMIBGeneralObjects.

apSysMgmtGeneralObjects Table (1.3.6.1.4.1.9148.3.2.1.1)

Object Name	Object OID	Description
apSysCurrentEndptsDenied	1.3.6.1.4.1.9148.3.2.1.1.26	Global counter for current endpoints denied

The *system* HDR collection group contains a new reporting field, *Current Deny Entries Allocated*, which contains the value of the global endpoints denied counter.

Anonymous Message Processing

RTN 2332

This release brings X123 Lawful Intercept (LI) processing into conformance with current German, Austrian, and Swiss regulatory requirements regarding the intercept and replication of a SIP INVITE emanating from an *Anonymous* source. Such INVITES contain a *From* header similar to the following:

From: Anonymous <sip:JohnSmith@privacy.invalid>;tag=hyh8

Previous implementations supported the identification and replication of targeted calls that contained an *Anonymous* value in the SIP INVITE *From* header when the calling party was the subject of an LI warrant. However, when the called party—not the calling party—was the warrant subject, the replicated X2 CDC (Call Content Channel) sent to the X123 Mediation Function failed to identify the calling party, beyond providing the non-descriptive contents of the *From* header.

With this release, a new processing model can be selected to enable a best-effort attempt to identify anonymous, non-targeted, calling parties.

Processing Anonymous From-Headers

Processing of Anonymous *From* headers, when the originator of the SIP INVITE is not the target of a current LI warrant takes place when the SIP INVITE egresses the Session Director.

Anonymous messages are processed as follows:

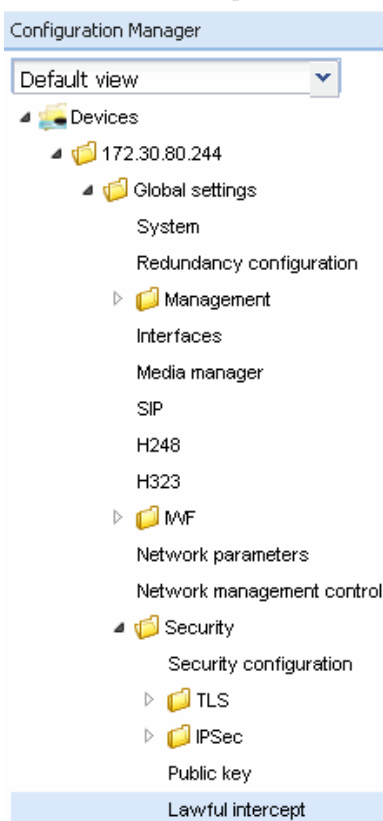
1. The Session Director examines the outgoing message for *P-Asserted-Identity* and *P-Preferred-Identity* headers. If such headers are found in the outgoing message, the header values are saved; if these headers are not present, no values can be saved.
2. The Session Director checks *Anonymous* entries in the registration cache to determine if the message originator is a registered endpoint. If *Anonymous* is a registered endpoint, the Session Director checks for *P-Associated-URI* entries that may have been provided by the Registrar. Such associated URIs may or may not be contained in the registration cache. If associated URIs are found in the cache, these URIs are saved; if associated URIs are not found, none can be saved.
3. For each identity found when parsing the outgoing SIP INVITE for *P-Asserted-Identity* and *P-Preferred-Identity* headers, and for each associated URI retrieved from the registration cache, the Session Director adds a *P-Asserted-Identity* header to the replicated X2 CDC sent to the X123 Mediation Function. Consequently the CDC sent the Mediation Function may not be identical to the SIP INVITE actually sent to the called party. Rather it can contain extra *P-Asserted-Identity* headers which convey the results of the Session Director's best effort to identify the *Anonymous* calling party. Obviously, if the parsing and retrieval operations fail to provide identification of the calling party, the best-effort fails, and the X2 CDC sent the X123 Mediation Function is identical to that of the egress SIP INVITE.

Enabling Regulatory-Compliant Processing of Anonymous From-Headers

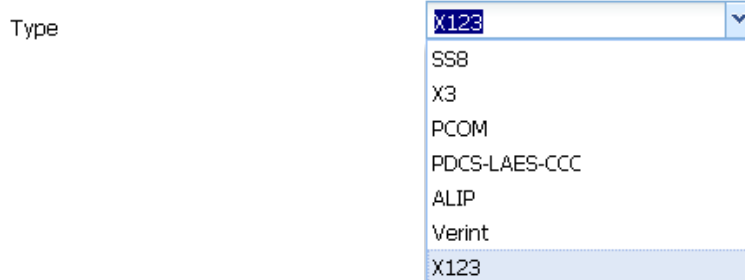
To enable regulatory-compliant processing of Anonymous From headers:

1. Expand the Configuration Manager slider > Global settings > Security.

- Click Lawful intercept.



- Type**—Click X123 in the drop down list.



- Scroll to X123.
- De-anonymize caller**—Click **enabled** or **disabled** in the drop down list. The default value is disabled.



- Click **Apply**. A Success message appears.
- Click **OK**.

250000 Trusted Endpoints

The Net-Net 4500 supports 250,000 Trusted endpoints as part of its DOS2 Security feature.

This feature is enabled automatically. There is no configuration required in Net-Net Central for this feature.

SBC Host Certificates SNMP

When a security certificate is installed locally on the Net-Net SBC, you can poll the expiration of the certificate using the **apSecurityCertificateTable**.

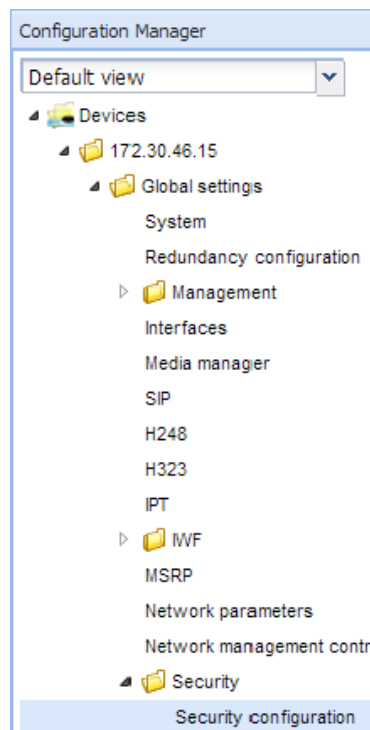
You can configure the Net-Net SBC to generate the **apSecurityCertExpiredNotification** trap once a certificate has expired. The number of minutes between notifications sent is configured in the *Security configuration* parameter **Local certificate trap interval**.

To send a warning of expiration, you can set the *Security configuration* parameter **Local certificate expiration warn period** to the number of days before the locally installed certificate expires in which you would like a warning.

Configuration Instructions

To configure the Net-Net SBC to generate traps when a certificate has or is about to expire:

1. Expand the Configuration Manager slider > Global settings > Security.



2. Click Security configuration.
3. **Local certificate expiration warn period**—Set the to the number of days before the locally installed certificate expires in order to receive a warning. A value of 0 disables the trap.
4. **Local certificate expiration trap period**—Set the to the number of minutes between notifications sent once a certificate has expired. A value of 0 disables the warning trap.
5. Click Apply.

Introduction

This section summarizes options for configuring the lawful intercept feature. It describes how the Net-Net SBC interoperates with mediation equipment from vendors who build lawful intercept equipment. If you are interested in the details of how this feature set works, refer to the *Net-Net LI Guide*.

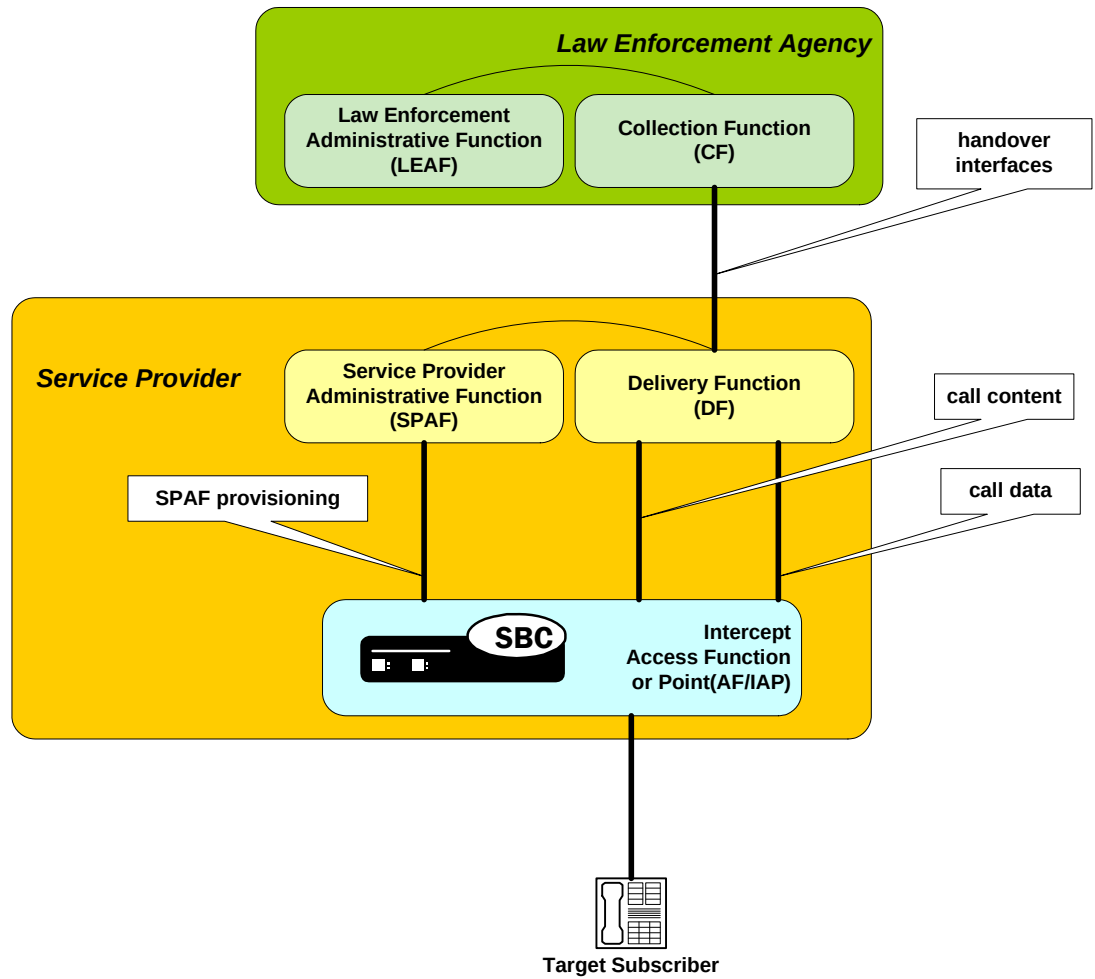
LI/CALEA consists of the interception of call content and/or the interception of call-identifying information. It requires that call information and media streams be sent to one or more law enforcement agencies in accordance with warrants for lawful interception.

You can configure your Net-Net SBC to support LI/CALEA functionality, enabling the Net-Net SBC to play a role in your Lawful Interception solution. Acting as an intercept access point (IAP), the Net-Net SBC can provide call data and can replicate media when signaling and media are anchored at the Net-Net SBC.

The Net-Net SBC supports LI/CALEA functionality that:

- Ensures unobtrusive intercept by hiding the network-based intercept of call information and content through topology hiding and media relay or NAT
- Intercepts and forwards call information and call content
- Interfaces with the mediation equipment [service provider administrative function (SPAF) and delivery function (DF)] for legal intercept

The following diagram provides one example of the Net-Net SBC deployed as part of a service provider's lawful intercept solution.



Recommendations

Calls may be lawfully intercepted by different devices in the service provider's network based on specific call flows, involvement of the device in the invoked service and where devices sit in the flow. Acme Packet recommends that you contact our Professional Services department to plan your use of the lawful intercept feature on your Net-Net SBC. Acme Packet Professional services can assist you with network/call flow analysis to determine which types of calls will involve the Net-Net SBC as an intercept access point and to recommend proper configuration.

Interoperability with SS8

The Net-Net SBC is configured to communicate with a trusted SS8 Xcipro SSDF for provisioning of target numbers by the SSDF and for delivery of call data (or call data and call content) by the Net-Net SBC to the SSDF.

The provisioning interface (INI-1) used between the Net-Net SBC and the SSDF is specified by SS8 in their Generic Interface (SS8 GI). Through this interface the Net-Net SBC is provisioned with target numbers and is informed if the lawful interception is for call data only, or call data and call content replication.

For the purpose of call identification, call data events that are delivered by the Net-Net SBC over the CDC (INI-2) to the SSDF. The interface used for CD between the Net-Net SBC and the SSDF is the SS8 GI specification, which identifies mapping of SIP messages to call data events to be delivered over the CD interface.

The Net-Net SBC can intercept the content of calls without the subscriber being able to detect any change and without introducing any additional latency. The Net-Net SBC can duplicate the content and deliver replicated media over the CCC (INI-3) to the SSDF. The interface used for CC between the Net-Net SBC and the SSDF is compliant with PacketCable specification PKT-SP-ESP-I03-040113 (encapsulation in UDP).

The DF routes the call data and call content to the law enforcement agency over standards-based handover interfaces to the Collection Function (CF).

Interoperability with Verint

The Net-Net SBC is configured to communicate with a trusted Verint Systems STAR-GATE platform for provisioning of target numbers by STAR-GATE and for delivery of call data (or call data and call content) by the Net-Net SBC to STAR-GATE.

The provisioning interface (INI-1) used between the Net-Net SBC and STAR-GATE is specified by Verint in their INI-1 specification. Through this interface the Net-Net SBC is provisioned with target numbers and is informed if the lawful interception is for call data only, or call data and call content replication.

For the purpose of call identification, call data events that are delivered by the Net-Net SBC over the CDC (INI-2) to STAR-GATE. The interface used for CDC between the Net-Net SBC and STAR-GATE is PacketCable PKT-SP-EM-I08-040113 (RADIUS/ UDP, with Verint extensions), which identifies mapping of SIP messages to call data events to be delivered over the CD interface.

The Net-Net SBC can intercept the content of calls without the subscriber being able to detect any change and without introducing any additional latency. The Net-Net SBC can duplicate the content and deliver replicated media over the CCC (INI-3) to STAR-GATE. The interface used for CC between the Net-Net SBC and STAR-GATE is compliant with PacketCable specification PKT-SP-ESP-I03-040113 (encapsulation in UDP).

The DF routes the call data and call content to the law enforcement agency over standard-based handover interfaces to the Collection Function (CF).

Interoperability Using a Dynamic Trigger by CMS

The Net-Net SBC is configured to communicate with a trusted CMS (call management server) to enable dynamic lawful interception through the use of an intercept trigger for a specific SIP call and the delivery of call content by the Net-Net SBC to the DF. In this model, the CMS provides call identifying information to the DF.

A dynamic trigger is used by the CMS to signal the Net-Net SBC to duplicate call content and to deliver the replicated content to a specified DF.

The Net-Net SBC can intercept the content of calls without the subscriber being able to detect any change and without introducing any additional latency. The Net-Net SBC can duplicate the content and deliver replicated media over the CCC (INI-3) to the specified DF. The interface used for CC between the Net-Net SBC and the specified DF must be compliant with PacketCable specification PKT-SP-ESP-I03-040113 (encapsulation in UDP).

The DF routes the call data (provided by the CMS) and call content (provided by the Net-Net SBC) to the law enforcement agency over standard-based handover interfaces to the Collection Function (CF).

Interoperability Using ALIP

The Net-Net SBC supports a lawful interface called Acme Packet Lawful Intercept Provisioning (ALIP).

The Net-Net SBC is configured to communicate with a trusted mediation device to enable dynamic lawful interception through the use of an intercept trigger for a specific SIP call and the delivery of call content by the Net-Net SBC to the DF. In this model, device provides call identifying information to the DF.

A dynamic trigger is used by the mediation equipment to signal the Net-Net SBC to duplicate call content and to deliver the replicated content to a specified DF.

The Net-Net SBC can intercept the content of calls without the subscriber being able to detect any change and without introducing any additional latency. The Net-Net SBC can duplicate the content and deliver replicated media over the CCC (INI-3) to the specified DF. The interface used for CC between the Net-Net SBC and the specified DF must be compliant with PacketCable specification PKT-SP-ESP-I03-040113 (encapsulation in UDP).

The DF routes the call data (provided by another device) and call content (provided by the Net-Net SBC) to the law enforcement agency over standard-based handover interfaces to the Collection Function (CF).

SS8 Xcipio Global Configuration

This section contains information about how to configure SS8 Xcipio LI support on your Net-Net SBC. You must enable LI administrative privileges in order to perform specific SS8 Xcipio configuration tasks. Without these privileges, you will not be able to view or access any of the LI configuration attributes.

Before You Configure

Before you configure your Net-Net SBC for lawful intercept deployment in an SS8 Xcipio environment, ensure that you have the following:

- The appropriate software license activated on the Net-Net SBC
- A network diagram that provides local and remote IP address and port information for the INI1 interface, and local address and port information for the INI2 and INI3 interfaces.

Note that you need not provide remote addressing information for the INI2 and INI3 interfaces. These Xcipio addresses are passed to the AF as part of the initial provisioning process described in *INI Interface* in Chapter 1, *SS8 Xcipio LI Interoperability*.

- Lists of the SIP headers you want to search: the first list containing one or more SIP headers (for example, *From* or *P-Asserted-Identity*) that identify the *calling* party; a second list containing one or more SIP headers (for example, *To* or *Request-URI*) that identify the called party; and a final list containing one or more SIP headers (for example, *Diversion* or *Referred-By*) that identify the *redirecting* party.

Configuring SS8 Xcipio Attributes

You configure global SS8 Xcipio attributes that:

- Name the SS8 Xcipio LI instance
- Provide a shared secret for authentication operations
- Specify the match criteria applied to incoming SIP INVITE request if the candidate call is subject to LI
- Specify IP addresses contained in replicated call content

You first assign a name (an ASCII string) to the SS8 Xcipio LI instance. The *name* parameter provides a handle used to distinguish multiple SS8 Xcipio LI instances from one another.

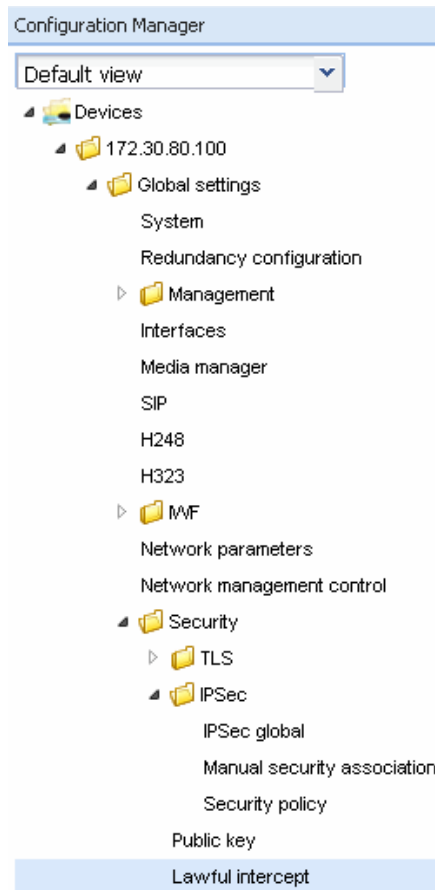
The **Authentication serial number** parameter (an ASCII string containing a minimum of 4 and a maximum of 32 alphanumeric characters) provides a shared secret known to both SS8 Xcipio and the Net-Net SBC, and is used for client/server authentication. This same value must be configured on both the Net-Net SBC and on SS8 Xcipio.

Enabling LI

You must enable the LI configuration prior to performing any SS8 Xcipio configuration.

To enable LI:

1. Expand the Configuration Manager slider > Global settings > Security.



2. Click Lawful intercept.
3. State—Select enabled from the drop-down list.

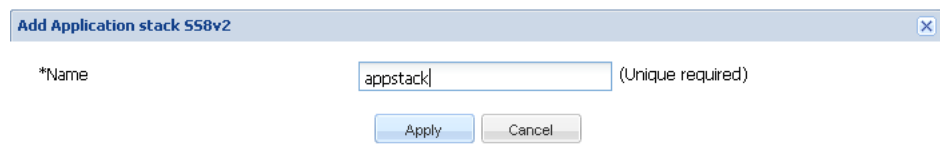
State

4. Click Apply.

Naming the SS8 Xcipio LI Instance

To configure the name of the SS8 Xcipio LI instance:

5. In the Lawful Intercept configuration screen, scroll to Application Stack SS8v2.
6. Click Add. The Add Application Stack Ss8v2 dialog box appears.
7. **Name**—Enter the name of the SS8 Xcipio LI instance.



8. Click Apply. The name appears in the Application Stack SS8v2 table.

Configuring the Shared Secret

To configure the shared secret:

9. In the Lawful Intercept configuration screen, scroll to Application Stack SS8v2.
10. Click the name of the SS8 Xcipio LI instance in the Application Stack SS8v2 table and click **Edit**.
11. Click **Edit** next to **Authentication serial number**. The Edit Encrypted data dialog box appears.
12. **New value**—Enter the name of the password, or shared secret, known to both SS8 Xcipio and the Net-Net SBC. This is an alphanumeric string of at least four characters, with a maximum of 32 characters.
13. **Confirm new value**—Re-enter the shared secret entered in step 6.
14. **Configuration password**—Enter the configuration password used to encrypt the configuration information on the Net-Net SBC.
15. **Confirm configuration password**—Re-enter the configuration password to confirm the password.

Authentication serial number

Edit

Edit encryption password

Enter new value

Confirm new value

Enter configuration password

Confirm configuration password

16. Click **OK**. A message appears recommending you run the ACLI command `verify-config` on the Net-Net SBC after you save your configuration.
17. Click **OK**.
18. Click **Apply**.

INI1 Interface Configuration

The INI1 interface supports the transfer of warrant information between the SS8 Xcipio SPAF (in the client role) and the Net-Net SBC (in the server role).

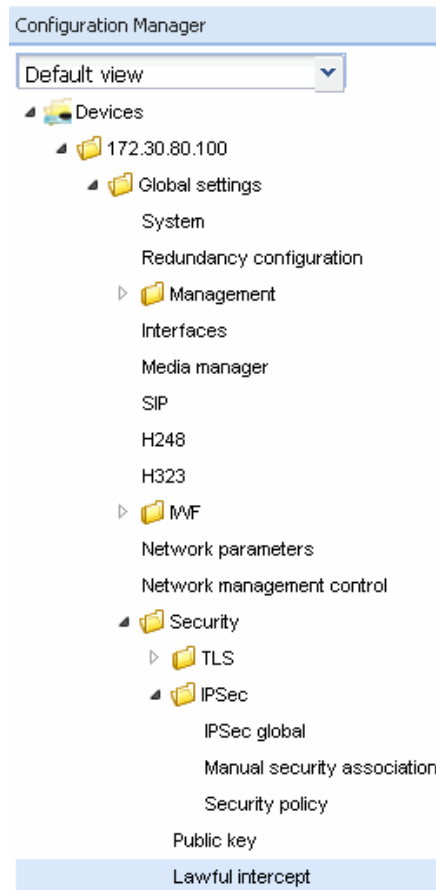
For the INI1 interface, you configure attributes that identify/specify the:

- Name of the INI11 interface instance
- INI1 realm
- IP address on which the Net-Net SBC listens for incoming INI1 client requests
- IP address and port number of the SPAF

Configuring INI1 Interface

To configure the INI1 interface:

1. Expand the Configuration Manager slider > Global settings > Security.



2. Click Lawful intercept.
3. Scroll to Application Stack SS8v2.
4. Select an SS8 Xcpio LI instance in the Application Stack SS8v2 table and click **Edit**.
5. Scroll to INI1 Agent.
6. Click **Add**. The Add INI1 Agent dialog box appears.
7. **Name**—Enter a name for this INI1 interface instance. The name provides a handle used to distinguish multiple INI1 interface instances from one another.
8. Click **Apply**.
9. Select the INI1 agent from the table and click **Edit**.
10. **Realm**—Select a realm name from the combination box to identify the realm that contains the INI1 interface specified by the **Source IP address**.
11. **Source IP address**—Enter an IP address to identify the local IP address monitored by the Net-Net SBC INI1 server for incoming INI1 client requests.

12. **Destination IP address and port**—Enter an IP address and port pair to identify the client IP address and port pair.

*Name	INI1agent
Realm	rc2 ▼
Source IP address	100.12.30.46
Destination IP address	100.10.15.20:80

13. Click **Apply**.
14. Click **Apply**.

INI2 Interface Configuration

The INI2 interface supports the transfer of call data messages between the Net-Net SBC and the SS8 Xcipio DF.

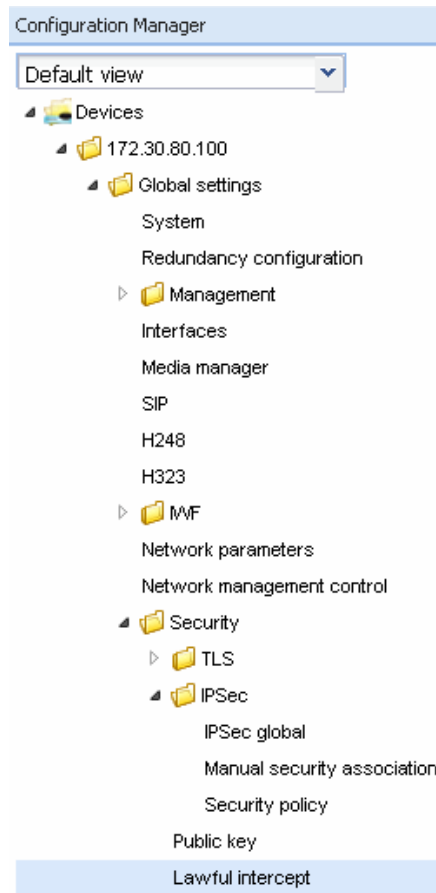
For the INI2 interface, you configure attributes that provide the:

- Name of the INI2 interface instance
- INI2 realm
- IP address of the local INI2 interface
- Amount of time that undeliverable call data messages are buffered before discard

Configuring INI2 Interface

To configure the INI2 interface:

1. Expand the Configuration Manager slider > Global settings > Security.



2. Click Lawful intercept.
3. Scroll to Application Stack SS8v2.
4. Select an SS8 Xcpio LI instance in the Application Stack SS8v2 table and click **Edit**.
5. Click **Add**. The Add INI2 Agent dialog box appears.
6. **Name**—Enter a name for this INI2 interface instance. The name provides a handle used to distinguish multiple INI2 interface instances from one another.
7. Click **Apply**.
8. Select the INI2 agent from the table and click **Edit**.
9. **Max buffering timer (sec) (Optional)**—Enter the number of seconds to set the INI2 buffer timers. The default value is 0 seconds and the maximum number is 30 seconds. A 0 value disables the INI2 buffer timer, meaning that no messages are buffered. With this timer set, the Net-Net SBC buffers call data messages for the specified time period in the event that the SS8 Xcpio DF is busy, down, or disconnected.
10. **Realm**—Click a realm name in the dropdown list to identify the realm that contains the INI2 interface specified by the **Source IP address**.

11. **Source IP address**—Enter an IP address to identify the local IP address monitored by the Net-Net SBC INI2 server for incoming INI2 client requests.

*Name	INI2agent
Maximum buffer time	0
Realm	▼
Source IP address	

12. Click **Apply**.
13. Click **Apply**.

INI3 Interface Configuration

The INI3 interface supports the transfer of call content from the Net-Net SBC to the SS8 Xcipio DF.

For the INI3 interface, you configure attributes that provide the:

- Name of the INI3 interface instance
- INI3 realm
- IP address of the local INI3 interface
- Amount of time that undeliverable call data messages can be buffered before discard

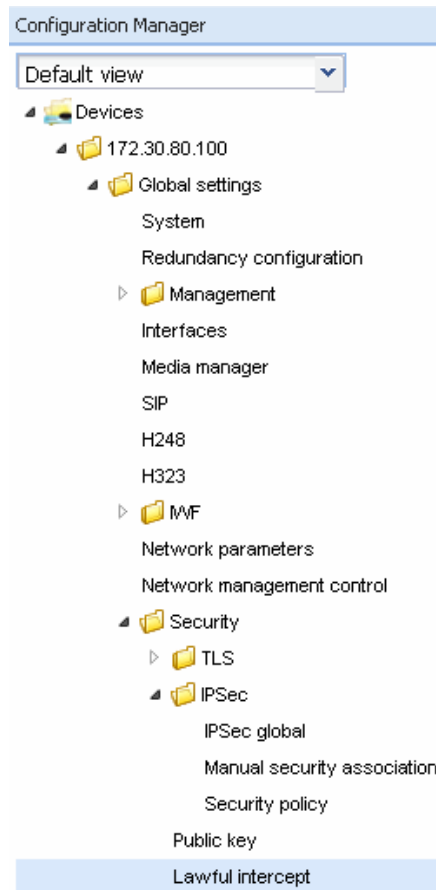
Enabling a Directional CCC-ID

Each replicated media stream has a Call Content Connection identifier (**Directional CCC ID**) that facilitates the correlation of associated data/signalling and media streams. You can configure whether or not the **Directional CCC ID** is directional, meaning that it indicates not only the media stream's association, but also the stream's directionality.

Configuring INI3 Interface

To configure the INI3 interface:

1. Expand the Configuration Manager slider > Global settings > Security.



2. Click Lawful intercept.
3. Scroll to Application Stack SS8v2.
4. Select an SS8 Xcpio LI instance in the Application Stack SS8v2 table and click **Edit**.
5. Click **Add**. The Add INI3 Agent dialog box appears.
6. **Name**—Enter a name for this INI3 interface instance. The name provides a handle used to distinguish multiple INI3 interface instances from one another.
7. Click **Apply**.
8. Select the INI3 agent from the table and click **Edit**.
9. **Directional CCC id (Optional)**—Click **enabled** or **disabled** from the dropdown list to enable or disable a Directional CCC ID. The default value (**disabled**) specifies the use of a non-directional CCC ID.
10. **Realm**—Click a realm name from the dropdown list to identify the realm that contains the INI3 interface specified by the **Source IP address**.

11. **Source IP address**—Enter an IP address to identify the local IP address monitored by the Net-Net SBC INI3 server for incoming INI3 client requests.

*Name	ini3agent
Directional CCC ID	enabled ▼
Realm	rc1 ▼
Source IP address	100.10.20.30:80

12. Click **Apply**.

LI Pre-Provisioned Targets to 30K

There is no configuration required for this feature. It is enabled automatically.

P-DCS-LAES: SIP Session Interception Mode

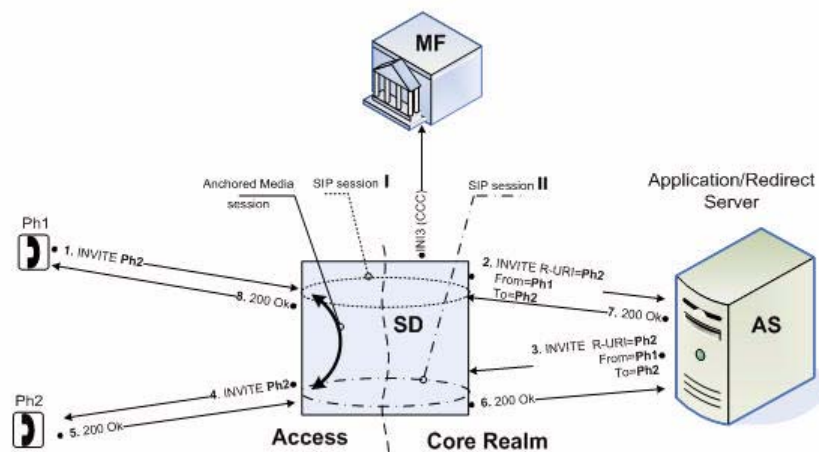
By default, hairpin (or spiral) calls that traverse the Net-Net SBC are treated as a single call, or single SIP session. That is, all individual call legs are intercepted/replicated as a single call (with a single ccc-id). The relation between call legs is determined using the SDP (media IP addresses) of each call-leg. This default mode is referred to as *single-session-mode*.

This default behavior can be changed so that no association between individual call legs is made. That is each call leg is treated as a single SIP session, and each call leg is intercepted/replicated with its own ccc-id. This mode is referred to as *all-sessions-mode*.

The new mode offers two advantages.

- Simplicity of implementation/detection of call topology changes after initial dialog establishment
- SDP need not be present in the first INVITE message as no relation between multiple call legs need to be established using SDP

With regard to Figure 1, the P-DCS-LAES header requesting intercept can appear in either of the SIP INVITE requests (messages 1 and 3), and/or in either of the INVITE responses (messages 5 and 7).

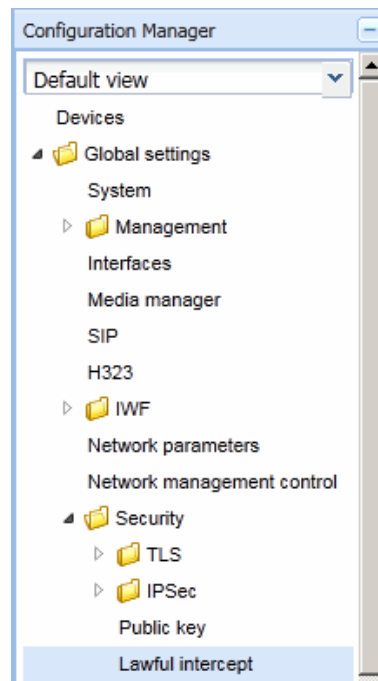


Assuming only one P-DCS-LAES header, the default *single-session-mode* reports a single instance of CCC data that replicates the contents of both SIP session I and SIP session II. By contrast, *all-sessions-mode* reports two instances of CCC data — the first instance reporting SIP session I content, and the second instance reporting SIP session II content.

Configuring Hairpin Reporting Algorithm

To specify the hairpin reporting algorithm:

1. Expand the Configuration Manager slider -> Global settings -> Security.



2. Click Lawful intercept.
3. Scroll to the Application stack PDCS-LAESv2 table and click **Add**.
4. **Name**— Enter a name for the PDCS-LAESv2 Application stack and click **Apply**.

*Name

5. Select the application stack that you created and click **Edit**.
6. **SIP session intercept mode**— Click **report-as-single-sessions** (default value) from the drop down list to provide a single instance of CCC data to the DF.
Click **report-all-sessions** from the drop down list to provide SIP-session-specific CCC data streams.

7. Click **Apply**.

Multiple Header Mode

Multiple Header Mode (available in S-C6.1.0 and subsequent releases) enables and manages the use of multiple P-DCS-LAES, which can be included within SIP INVITE request and response messages.

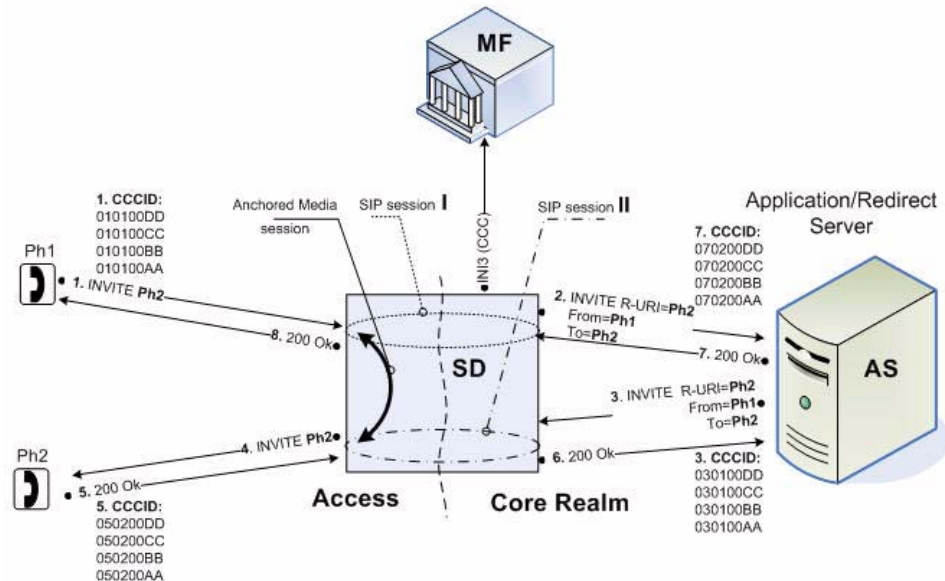
With Multiple Header Mode enabled, two simple rules are used to limit the maximum number of P-DCS-LAES headers subject to examination.

- A maximum of three P-DCS-LAES headers are examined per message (INVITE request or response).
- A maximum of four P-DCS-LAES headers are examined per SIP session.

Any additional headers are silently discarded; consequently, a media flow can be replicated a maximum of 8 times.

The **multiple header mode** attribute specifies the processing algorithm for multiple P-DCS-LAES headers. Valid values for **multiple header mode**, are **override-with-most-recent** (the backward compatible default value) and **accumulate-all**, which is only available when **SIP session intercept mode** is set to **report-all**.

The following illustration and table are used to illustrate the alternate processing algorithms.



The illustration shows a call sequence with 4 P-DCS-LAES headers, with corresponding CCCIDs, in the messages 1 and 3 (INVITE requests), and messages 5 and 7 (INVITE responses). Note that the CCCIDs are patterned — the first 3 digits indicate the message number, the second 3 digits indicate the message type (100 for INVITE request, 200 for INVITE response.), and the last two hexadecimal digits indicate relative position (order of appearance) when they appear in the same message.

Because previously described rules set hard limits on the number of P-DCS-LAES headers examined per INVITE request or response, and per SIP session, some of the headers (depicted by unique CCCIDs in Figure 8) are silently discarded. The behavior is described in the sections below based on the configuration settings.

override-with-most-recent

is the default (backward compatible) value for the **multiple header mode** attribute. It is available regardless of the value assigned to the **SIP session intercept mode** attribute. With regard to Figure 8 only a single P-DCS-LAES interception occurs with the CCCID 070200DD.

When **SIP session intercept mode** is set at **combine-spiral-sessions** (the default value), the algorithm is applied as follows.

1. Examine each INVITE request and response message and keep only the top header — leaving 010100DD at the end of processing message 1

2. Provision the corresponding media interception
3. Apply the same rules to message 3 — resulting in a new CCCID 030100DD which replaces the previous one from message 1
4. Apply the same rules to messages 5 and 7 — resulting in the final CCCID 070200DD, which is then used for programming the media interception packet header.

When **SIP session intercept mode** is **report-all-sessions**, the algorithm is applied as follows.

1. Examine each message and keep only the top header — leaving 010100DD (SIP session I) at the end of processing message 1
2. Provision the corresponding media interception
3. Apply the same rules to message 3 — resulting in 030100DD (SIP session II) at the end of processing message 3
4. Apply the same rules to message 5 — resulting in the final CCCID 050200DD, (SIP session 2) which is then used for programming the media interception packet header.
5. Apply the same rules to message 7 — resulting in the final CCCID 070200DD (SIP session I), which is then used for programming the media interception packet header.

accumulate-all

is available only when **SIP session intercept mode** is set to **report-all-sessions**. It enables multiple media interception of the same SIP session triggered by multiple headers present in SIP INVITE request and responses. As stated above, we have hard limits for P-DCS-LAES headers: 3/message, 4/session.

The algorithm is applied as follows.

1. Parse and store the top 3 P-DCS-LAES headers. Silently discard additional headers following the 3 header/message rule. After processing message 1, the list for SIP session I contains: 010100DD, 010100CC, and 010100BB.
2. Similarly, after processing message 3, the list for SIP session II contains 030100DD, 030100CC, and 030100BB.
3. Parse message 5 and extract the top 3 headers — as a result hitting the 4 header /session limit for session II. Based on a *first-come-first-served* algorithm, retain the 3 existing headers in session SIP II and append the top header from message 5 to the list. The final header list for SIP session II contains 030100DD, 030100CC, 030100BB, and 050200DD.

- Using the same logic parse message 7 as above and the resulting SIP session I list will contains 010100DD, 010100CC, 010100BB, and 070200DD — resulting in a total of 4 interceptions for session I and 4 interceptions for session II.

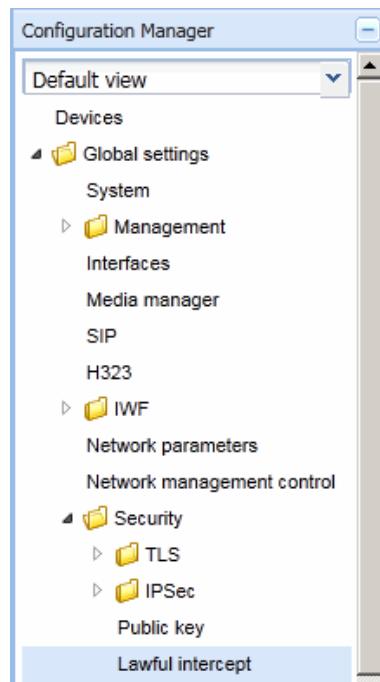
The following table summarizes the results:

SIP session intercept mode: Multiple header mode:	combine-spiral-sessions override-with-most-recent	report-all override-with-most-recent	report-all accumulate-all
Media Interception(s) with ccc-id	070200DD	050200DD 070200DD	010100DD 010100CC 010100BB 070200DD 030100DD 030100CC 030100BB 050200DD

Configuring Multiple Header Mode Algorithm

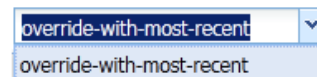
To specify the multiple header mode algorithm:

- Expand the Configuration Manager slider -> Global settings -> Security.



- Click Lawful intercept.
- Multiple header mode** — Click one of the following choices:
 - override-with-most-recent** (default value) to choose it from the drop down list to limit examination to a single P-DCS-LAES header per call-leg (SIP session).
 - accumulate-all** to choose it from the drop down list to examine up to 4 P-DCS-LAES headers per call-leg (SIP session).

Multiple header mode



4. Click **Apply**.

Introduction

The Common Open Policy Service (COPS) [RFC 2748] is a protocol supported by the Net-Net SBC to perform and implement Call Admission Control (CAC) based on the policies hosted in an external policy server. While the Net-Net SBC already supports internal CAC policies, they are not as flexible as a Resource and Admission Control Function / Policy Decision Function (RACF/PDF), the generic resource and admission control functional architecture conceived by the ITU-T and the IETF.

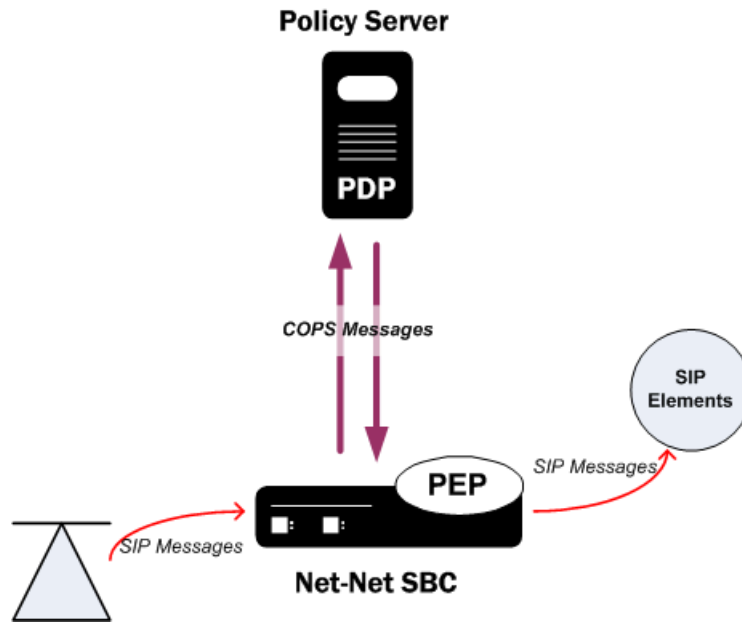
The Net-Net SBC COPS model includes a Policy server, functionally called the policy decision point (PDP), and the edge router, functionally called the policy enforcement point (PEP), the Net-Net SBC itself. The PDP and the PEP communicate with each other via the COPS protocol.

The Net-Net SBC also supports CLF services with its COPS implementation. Although the purpose of CLF is unlike the RACF/PDP functionality, COPS is the protocol the Net-Net SBC uses to talk to a CLF network device.

One of three licenses is required to use External Policy Server services: External Bandwidth Management-for RACF support.; External CLF Mgmt-for CLF support; External Policy Services-for support of both RACF and CLF.

Call Admission Control

Admission control is performed according to the following typical scenario. When the Net-Net SBC receives a SIP INVITE, it sends a COPS request (REQ) message to the PDP. The REQ message includes the call ID, the SIP client's IP address, the Net-Net SBC's IP address and port number of the ingress interface for the call, and SDP based bandwidth requirements. The PDP responds with a COPS Decision (DEC) message with either the Install or Remove command. An Install command directs the Net-Net SBC to forward the INVITE to the next SIP device. A Remove command directs the Net-Net SBC send a SIP 503 Service Unavailable message sent back to the UA and reject the call.



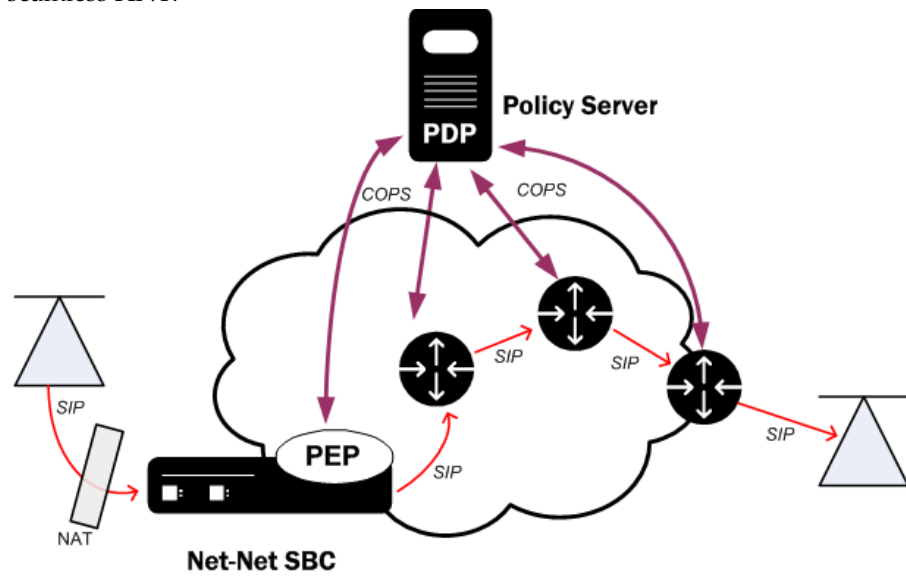
The Net-Net SBC can be configured so that both sides of a call, based on realm, are subject to COPS bandwidth enforcement. Each flow is treated as a unique call/event, because from a media and signaling perspective, they are distinct. As the Net-Net SBC functions as one side of a call, its IP address is inserted into the REQ message regardless of whether it is the calling or called party. This allows for the COPS install or remove decision to be made before the Net-Net SBC receives the 200 OK response, and before ringing the far-end phone. Only one external policy server can be used within a single realm.

When a call ends, either with the expected SIP BYE or CANCEL methods, or due to other error conditions, the Net-Net SBC will delete the reservation on the PDP by sending a COPS delete request state (DRQ) message to the PDP. All ended calls must be deleted from the PDP in order to accurately track used and available bandwidth.

Implementation Features

As the Net-Net SBC proxies and forwards calls, caller media information is known before the INVITE reaches the callee. The PEP can request a specific amount of bandwidth for a call, and the PDP can reserve this amount of bandwidth for a call before the called phone rings. A call's required bandwidth can also be reserved by network devices along the path from the caller to callee if the QoS admission criteria is pushed to PEPs such as routers, along this path to the callee.

The RACF can apply its hosted policies for calls originating at SIP UAs located behind NATs. This is a standard part of the Net-Net SBC's ability to provide seamless HNT.



Bandwidth Negotiation

Because the decision whether to admit or reject a call is made before the INVITE is forwarded to the called party, some information is not available to the PDP at the initial request. The final IP Address, UDP port number, that transport the RTP flow, and the codec used are not known by the Net-Net SBC until the called party responds with its own SDP information (either in the 180 or 200 response).

The Session Director sends a request to the PDP requesting as much bandwidth as the codec with the highest bandwidth in the SDP message requires. If the call is admitted, and when the called party returns finalized SDP information, the Session Director will modify the original reservation with the chosen codec's bandwidth requirements. This ensures the PDP has current and accurate information with which to make policy decisions.

COPS Connection

The COPS session is established over a persistent TCP connection between the PDP and PEP. A COPS Client-Open (OPN) message is sent from the Net-Net SBC to the RACF, which responds with a COPS Client-Accept (CAT) message. A COPS Client-Close (CC) message is sent to either side to gracefully close the persistent connection. This COPS connection is expected to never close, unless an error occurs.

COPS Failures

Connection failures are discovered through a keep alive mechanism. Keep alive (KA) messages are periodically sent by the Net-Net SBC to the RACF regardless if any other COPS messages have been exchanged. When a KA message is not received, a connection failure is flagged. If the COPS connection fails, the Net-Net SBC will continually try to re-establish the connection to the PDP. Previously established calls will continue unaffected, but the Net-Net SBC will deny new calls from being established until the CIPS connection is restored.

Failure Detection

A COPS connection failure is triggered by one of the three following events:

1. COPS KA timeout. The Net-Net SBC flags a COPS KA timeout when it does not receive a response for the KA it sent to the PDP. The PDP flags a COPS KA timeout when it does not receive the KA message within its requested timer time from the Net-Net SBC. At a minimum, when the COPS KA message times out, the TCP socket is closed.
2. Explicit COPS CC. The Net-Net SBC closes a COPS connection if it receives a COPS CC message from the PDP. The PDP closes a COPS connection if it receives a CC message from the Net-Net SBC. After the COPS layer connection is closed, then the TCP socket is closed too.
3. TCP socket termination. If either side receives a TCP FIN or RST, the TCP socket closes as expected. The COPS layer then detects that the socket has been closed before sending any further messages, and thus the COPS connection is closed.

Failure Recovery

The Net-Net SBC assumes that the PDP has a mechanism that re-uses the same logical IP Address, restarts itself in a timely manner, or fails over to another PDP. Therefore, no backup PDP IP address is configured on the Net-Net SBC.

The Net-Net SBC will try to re-open the COPS connection to recover from a connection failure. The PDP is never the device to initiate a connection. The Net-Net SBC increases its retry interval after successive reconnect failures. Once the retry interval has grown to every five minutes, the Net-Net SBC continues to retry to open the COPS connection at the five minute interval.

Net-Net High Availability Support for COPS

The Net-Net SBC's high availability (HA) capabilities have been extended to support COPS. When one Net-Net SBC in an HA configuration goes down, the MAC addresses are reassigned to a healthy Net-Net SBC. IP addresses "follow" the MAC addresses to provide a seamless switchover between HA nodes.

After an HA failover, the COPS connection on the primary Net-Net SBC is either gracefully torn down, or times out depending on behavior of the PDP. The backup Net-Net SBC attempts to create a new COPS connection with the PDP. The OPN message uses the same PEPID and Client Type as in the previous pre-failover session.

COPS Debugging

A new argument has been added to the show command for viewing COPS and CAC statistics. From the user prompt, type **show <space> ext-band-mgr <return>**.

ACMEPACKET# **show ext-band-mgr**

10:11:38-194

EBM Status	Active	High	Total	Total	PerMax	High
Client Trans	0	0	0	0	0	0
Server Trans	0	0	0	0	0	0
Sockets	1	1	1	1	1	1
Connections	0	0	0	0	0	0

	Recent	Total	PerMax
Reserve	0	0	0
Modify	0	0	0
Commit	0	0	0
Remove	0	0	0
EBM Requests	0	0	0
EBM Installs	0	0	0
EBM Errors	0	0	0
EBM Rejects	0	0	0
EBM Expires	0	0	0
EBMD Errors	0	0	0

You can also refer to the `log.ebmd` log file located in the `/ramdrv/logs/` directory on the Net-Net SBC. This file must be retrieved via FTP or SFTP.

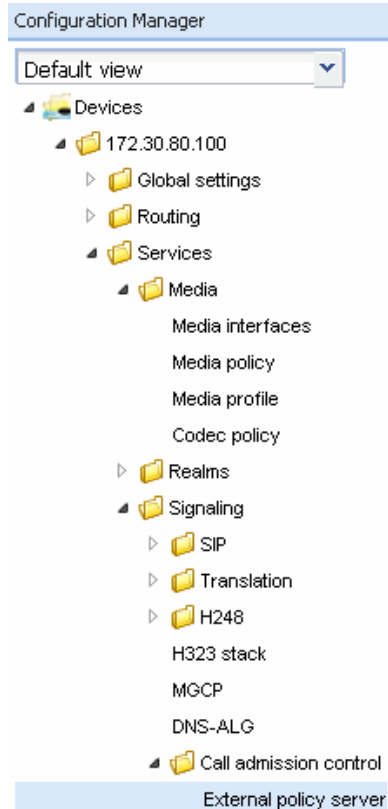
Configuring COPS

This section describes how to configure a COPS CAC scenario using Net-Net Central. In the following configuration examples, we assume that your baseline configuration passes SIP traffic, with the Net-Net SBC in the role of an Access SBC. In this example, you will configure additions to the realm configuration and the new external bandwidth manager configuration. You must also configure media profiles to accept bandwidth policing parameters.

Creating a Policy Server

To create a policy server:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click **Add** to define and create a new external policy server. The External Policy Server window appears.
4. **Name**—Enter the name of this CAC server.
5. Click **Apply**.
6. Select the new external policy server in the table and click **Edit**.
7. **State**—Select enabled from the combination box to enable this server.
8. **Operation mode**—Select bandwidth-mgmt in the Operational mode combination box.
9. **Policy communication protocol**—Select COPS in the Policy communication protocol combination box.

10. **IP address**—Enter the IP address of the CAC server in the IP address field.
11. **Port number**—Enter the port number that the COPS connection connects to on the CAC server.
12. **Realm in which server resides**—Select the realm in which this PDP exists from the combination box.
13. **Realm location**—Set this parameter to configure the child realm or its parent for the Address-Realm in the Globally-Unique-Address AVL in DIAMETER UDR messages that the Net-Net SBC sends to the policy server.
14. **Number of policy connections**—Enter the number of policy protocol TCP connection to establish to the PDP.
15. **Early resource reservation**—Select enabled from the combination box to enable early resource reservation.

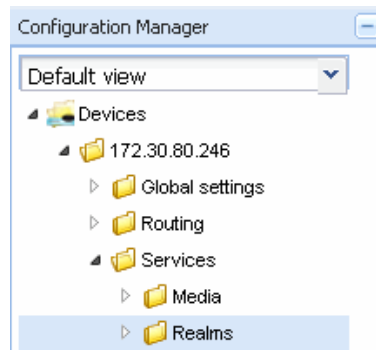
State	enabled
*Name	test
Operation mode	bandwidth-mgmt
Policy communication protocol	COPS
IP address	172.1.50.30
Port number	80
Realm in which server resides	svtest-access
Realm location	realm-in
Number of policy connections	10
Early resource reservation	enabled
Permit call when connection down	disabled
Set proxy server flag	enabled

16. **Permit call when connection down**—Select enabled from the combination box to enable the Net-Net SBC's ability to permit calls if there is no connection to the external policy server.
17. **Set proxy server flag**—Select enabled from the combination box if you want to the proxy bit in the header.
18. Click **Apply**. The External Policy Server window appears with the newly created policy server.

Configuring Realms

To configure realms:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to configure and click **Edit**.
4. Scroll to Manage media between endpoints.
5. **Within this realm serviced by this Net-Net system only**—Select enabled from the combination box to enable calls from devices in the same realm to have their media flow through the Net-Net SBC and be subject to COPS CAC.
6. **Within different realms in the same subnet serviced by this Net-Net system only**—Select enabled from the combination box to steer all media traveling between two endpoints located in different realms, but within the same network. If this field is set to disabled, then each endpoint will send its media directly to the other endpoint located in a different realm, but within the same network.
7. **Within this realm serviced by this Net-Net system only**—Select enabled from the combination box to steer all media traveling between two endpoints located in the same realm through the Net-Net SBC, to be subjected to COPS CAC..

Manage media between endpoints

Within this realm serviced by this Net-Net system only	enabled
Within different realms in same subnet serviced by this Net-Net system only	enabled
Within this realm serviced by different Net-Net systems (SIP only)	disabled

8. Scroll to Resource Admission Control Function (RACF).
9. **External policy server**—Select the external policy server you created in Step 3 in the External policy server drop down box.

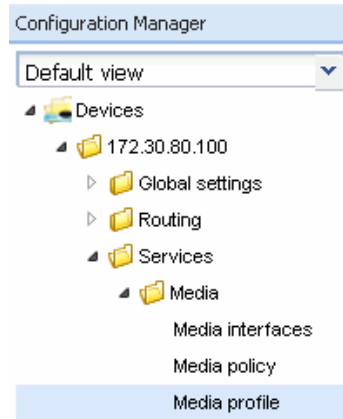
External policy server	test

10. Click **Apply**.

Configuring Profiles

To configure profiles:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
 3. Select the media profile you want to configure and click **Edit**.
 4. **Peak rate limit (bytes/sec)**—Enter the r, P (Token bucket rate / Peak data rate) value in this field.
 5. **Maximum burst size (bytes)**—Enter the b, m, M (Token bucket size / Min policed unit) value in this field.
- | | |
|------------------------------------|--------------------------------|
| Peak rate limit (bytes per second) | <input type="text" value="0"/> |
| Maximum burst size (bytes) | <input type="text" value="0"/> |
6. Click **Apply** to continue.

Diameter: Subscriber Information Enhancement

You can configure the Net-Net SBC to send the Subscription-Id AVP in the AAR message destined for policy servers. The Subscription-Id AVP contains the user's URI information, and is used in 3GPP Rx reference points. You might need to configure this functionality if your network's policy servers rely on having this information to identify endpoints/subscribers. Without this function enabled, the Net-Net SBC provides the L3 IP address, which is not enough information for some policy servers.

About the Subscription-Id AVP

The Subscription-Id AVP (code 443; type is Grouped) identifies the end-user's subscription and contains the following:

- Subscription-Id-Type AVP—(code 450; type is enumerated) Defines the subscription type as assigned by IANA; the server must implement all types required to perform credit authorization for the services it offers, including possible future values. Subscription-Id-Types must be created according to the "M" flag rule.
 - END_USER_E164 (0)—International E.164 format
 - END_USER_IMSI (1)—International IMSI format
 - END_USER_SIP_URI (2)—SIP URI format
 - END_USER_NAI (3)—Network Access Identifier format

- END_USER_PRIVATE (4)—Credit-control server private identifier
- Subscription-Id-Data AVP—(code 444; type is UTF8String) Identifies the end user. The Net-Net SBC populates this AVP with the subscriber's Request-URI corresponding to the AAR message. In other words, the Subscriber_id URI corresponds to the currently supported IP address of the subscriber included in the Frame-IP-address AVP.

If the Request-URI is of the type:

- Tel URI, the Subscription-Id-Type AVP is END_USER_E164
- SIP or SIPS URI, the Subscription-Id-Type AVP is END_USER_SIP_URI

Currently, the Net-Net SBC supports SIP, SIPS, and Tel URIs. So the Net-Net SBC sends out the Subscription-Id AVP only if the Request-URI received is one of these three types.

Configuring Subscription-Id AVP Support

To add Subscription-Id AVP support to your external policy server configuration:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click the external policy server you want to configure in the table and click **Edit**. The Edit external policy server window appears.
4. Click **Options** next to Optional AVPs. The Options dialog box appears.
5. Click **include-sub-info** in the Available options list to select it.

6. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

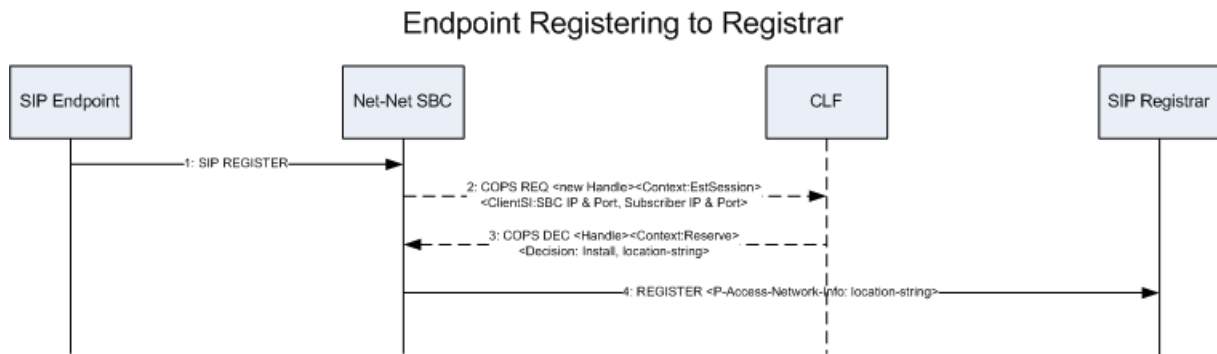
Connectivity Location Function

A Connectivity Location Function (CLF) maintains mappings between endpoints with dynamically assigned IP addresses and their physical location. The Net-Net SBC, acting as a P-CSCF, is the intermediary device between a registering endpoint and a CLF. The CLF thus validates and tags a registering endpoint, and the Net-Net SBC applies the CLF's actions. The Net-Net SBC and the CLF maintain a connection with each other using the COPS protocol.

CLF Behavior

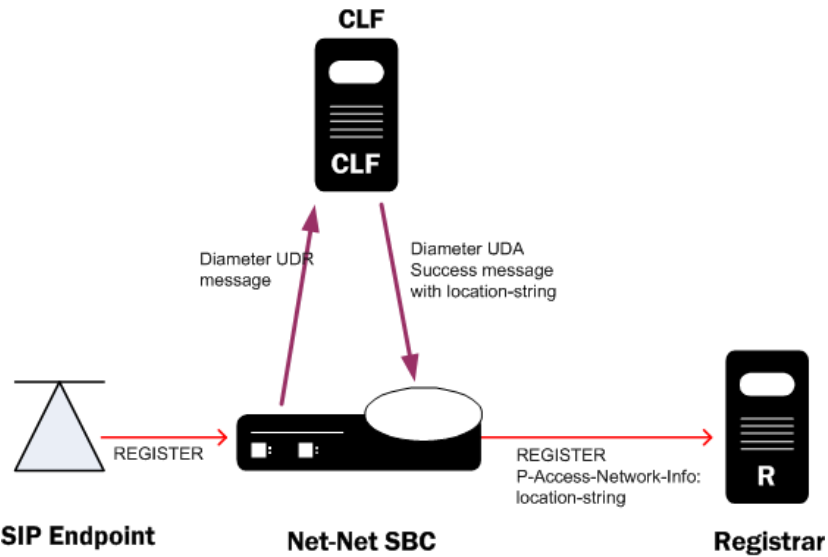
The Net-Net SBC and a CLF only interact when an endpoint registers or re-registers. The Net-Net SBC, acting as the P-CSCF, is the first SIP device that the REGISTER message reaches. Upon receiving the REGISTER message(1), the Net-Net SBC queries the CLF using the COPS protocol. The endpoint's (public) IP address and port, and the Net-Net SBC's IP information are sent to the CLF in a COPS REQ message(2).

The CLF responds to the Net-Net SBC with an Approve or Reject COPS DEC message(3). If the request is approved, then the CLF also sends a location-string value to be inserted in one of the SIP headers. The Net-Net SBC inserts a P-Access-Network-Info header containing the location-string into the incoming REGISTER message and forwards this message(4) to the SIP registrar/I/S-CSCF.



The Net-Net SBC will insert this P-Access-Network-Info header into all subsequent SIP messages from this endpoint as they are forwarded into the core network. The P-Access-Network-Info header is inserted into all SIP requests and responses except for ACK and CANCEL messages. For all boundaries where SIP messages pass from

trusted to untrusted SIP interfaces or session agents, the Net-Net SBC will strip out the P-Access-Network-Info header as expected.



If the CLF responds with a Reject DEC message, the Net-Net SBC rejects the registration, and sends a 503 - Service Unavailable message back to the registering endpoint. In this way, the CLF can be used for admission control.

The Net-Net SBC communicates with the CLF solely for retrieving location information from the CLF, and not for notifying the CLF about an endpoint's registration state or activity. When an endpoint's registration ends, either through a normal expiration, getting rejected by the registrar, or through specific de-registering or error conditions, the Net-Net SBC deletes the locally cached registration location string. The Net-Net SBC does not update the CLF about any registrations that have been deleted.

P-Access-Network-Info Header Handling

The P-Access-Network-Info header is created and populated according to the following rules:

1. If the CLF returns an Accept DEC message and a location string, the Net-Net SBC inserts the location string into a P-Access-Network-Info header in the outgoing REGISTER message.
2. If the CLF returns an Accept DEC message without a location string, the Net-Net SBC inserts the configured default string into a P-Access-Network-Info header in the outgoing REGISTER message.
3. If the CLF returns an Accept DEC message without a location string and no location string is configured on Net-Net SBC, the outgoing REGISTER message is forwarded out of the Net-Net SBC, but no P-Access-Network-Info header is created for the REGISTER message.

CLF Re-registration

The Net-Net SBC will send a new REQ message to the CLF to request a new location string if any of the following events occur:

1. The endpoint's contact address changes.
2. The SIP Register message's Call-ID header changes.
3. The endpoint's public IP Address or UDP port changes.
4. The endpoint connects to a different SIP interface, port, or realm on the Net-Net SBC than it did in the initial REGISTER message.
5. The registration expires in the Net-Net SBC's registration cache.

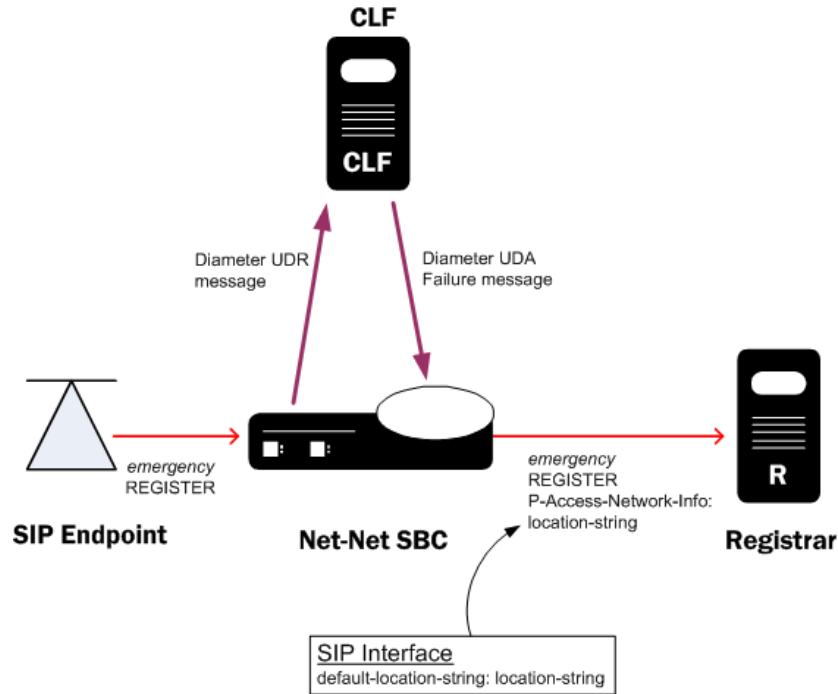
CLF Failures

If a COPS connection fails, the Net-Net SBC will continually try to re-establish the connection. Endpoints that are already registered will stay registered unless they timeout or if the registrar rejects their refreshes. When the COPS connection has not been established, and an endpoint registers on a SIP interface that is configured to use CLF, the Net-Net SBC forwards new REGISTER messages to the registrar using the default location string.

CLF Emergency Call Handling

The Net-Net SBC allows emergency calls into the network even if the endpoint that places the emergency call is not registered. In the expected fashion, the Net-Net SBC will query the CLF first for an incoming emergency call sourced from an unregistered endpoint. If the CLF response is successful, then the Net-Net SBC will insert the string returned from the CLF into a P-Access-Network-Info header, and insert this header into the emergency call's REGISTER message. If no location string is returned with a successful CLF response, the default location string is inserted into P-Access-Network-Info header.

If the CLF's response is to reject the emergency call, the Net-Net SBC will insert the configured default location string into the P-Access-Network-Info header and forward the emergency call's REGISTER message toward the registrar. For emergency calls where the endpoint has already successfully registered, the call will be routed into the network using the expected methods for emergency call routing.



If the COPS connection to the CLF is down, emergency calls from un-registered endpoints are still allowed into the network using the default string inserted into the emergency messages.

HA Functionality

The location strings generated by the CLF are replicated on the standby Net-Net SBC in an HA pair. This is required so that a Net-Net SBC in an HA pair can instantly continue processing calls using the previously learned CLF information.

CLF Debugging

A new argument has been added to the show command for viewing CLF statistics. From the user prompt, type **show <space> ext-clf-svr <return>**.

```
ACMEPACKET# show ext-clf-svr
14:17:14-114
EBM Status
```

	-- Period --			----- Lifetime -----		
	Active	High	Total	Total	PerMax	High
Client Trans	0	0	0	0	0	0
Server Trans	0	0	0	0	0	0
Sockets	0	0	0	0	0	0
Connections	0	0	0	0	0	0


```

          ----- Lifetime -----
          Recent      Total  PerMax
CLF Requests      0        0      0
CLF Admits        0        0      0
CLF Errors         0        0      0
CLF Rejects       0        0      0
CLF Expires       0        0      0
CLFD Errors       0        0      0

```

You can also refer to the `log.ebmd` log file located in the `/ramdrv/logs/` directory on the Net-Net SBC. This file must be retrieved via FTP or SFTP.

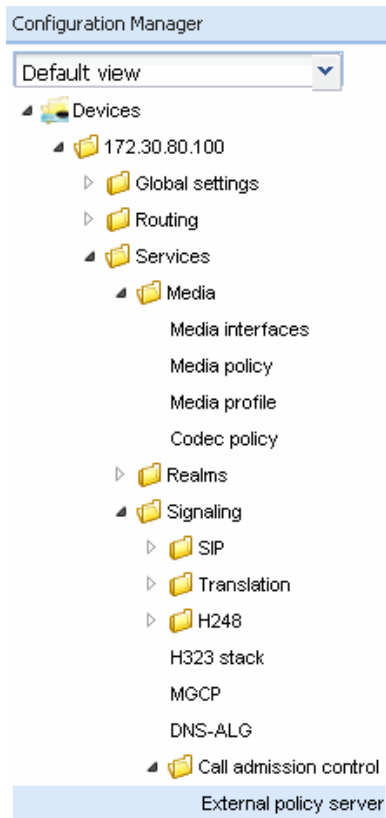
Configuring CLF

In the following configuration examples, we assume that your baseline configuration passes SIP traffic, with the Net-Net SBC in the role of an access SBC. In this example, you will configure additions to the realm configuration and the new external policy server configuration.

Configuring CLF

This section describes how to configure a COPS CLF scenario using the Net-Net Central.

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click **Add** to define and create a new external policy server. The External Policy Server window appears.
4. **Name**—Enter the name of the external policy server.
5. Click **Apply**.
6. Select the new external policy server in the table and click **Edit**.
7. **State**—Select enabled from the combination box to enable this server.
8. **Operation mode**—Select admission-control in the combination box.
9. **Policy communication protocol**—Select COPS in the combination box.
10. **IP address**—Enter the IP address of the CLF server.
11. **Port number**—Enter the port number that the COPS connection connects to on the CLF server.
12. **Realm in which this server resides**—Select the realm in which this CLF server exists from the combination box.
13. **Number of policy connections**—Enter the number of policy protocol TCP connection to establish to the CLF.
14. **Early resource reservation**—Select enabled from the combination box to enable early resource reservation.

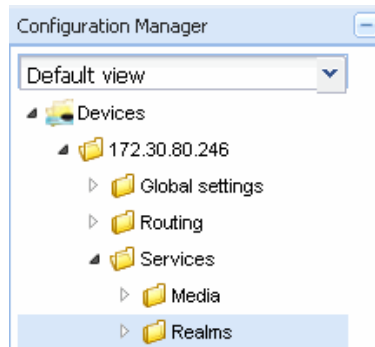
15. Click **OK**. The External Policy Server window appears with the newly created policy server.

State	enabled
*Name	test
Operation mode	admission-control
Policy communication protocol	COPS
IP address	192.168.50.40
Port number	80
Realm in which server resides	
Realm location	realm-in
Number of policy connections	1
Early resource reservation	enabled

Configuring the SIP Interface

To configure the SIP interface: Realm

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a realm that you want to configure and click **Edit**.
4. Select the Services/Protocols tab.
5. Select a configured SIP interface that will perform CLF.
6. Click **Edit**. The SIP interface window appears in the configuration pane, with the SIP Interface tab selected.
7. Scroll to Connectivity Session Location Function (CLF).
8. **Policy server**—Select the external policy server you created in Step 3 of the previous task in the Policy server drop down box.
9. **Default location string**—Enter the default location string in the Default location string field.

Connectivity Session Location Function (CLF)

Policy server	test ▼
Default location string	location-string

10. Click **Apply**.

Diameter: CLF and RACF

The Net-Net SBC supports Diameter base protocol (RFC 3588), which is used for Bandwidth-Based Call Admission Control (CAC) and Connectivity Location Function (CLF) applications. The existing licenses for COPS-based CLF and RACF support Diameter and COPS.

Diameter Connection

The Net-Net SBC supports Diameter (RFC 3588) connections to a Diameter server over TCP. The base Diameter protocol runs on TCP port 3868. Diameter-based CAC and CLF are available from the front media interfaces on the Net-Net SBC.

The Diameter connection is always initiated from the Net-Net SBC to the Diameter server. The Net-Net SBC begins the connection by sending a Capabilities-Exchange-Request (CER) to the server, which replies with Capabilities-Exchange-

Answer (CEA) message. If either side decides to close the persistent connection in a graceful manner, it can send a Disconnect-Peer-Request (DPR) message.

Diameter Heartbeat

Device-Watchdog-Request (DWR) and Device-Watchdog-Answer (DWA) messages are used to detect transport failures at the application layer between the Net-Net SBC communicating with a policy server via Diameter. The request/answer message pair forms a heartbeat mechanism that can alert the requesting side if the answering side is not reachable.

The Net-Net SBC always responds to a DWR by replying with a DWA message. In addition, the Net-Net SBC can be configured to initiate DWR messages toward a policy server or other Diameter-based network device.

You configure the **Keepalive message interval (sec)** parameter with a timeout value that determines the number of seconds a DWA is expected in response to the Net-Net SBC sending a DWR.

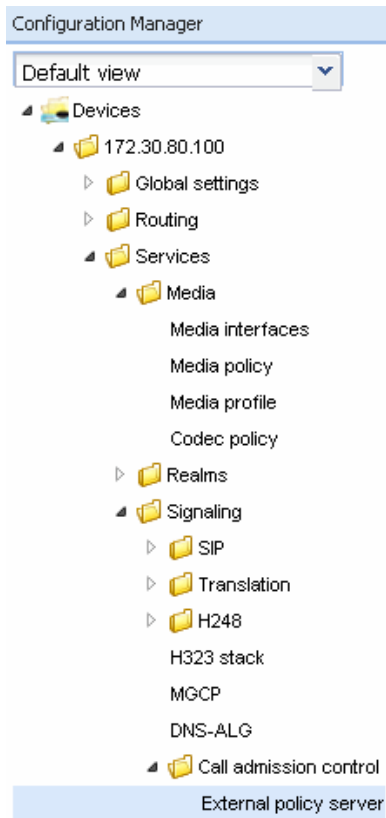
If the Net-Net SBC fails to receive a DWA response from a Policy Server within the configured interval, then the connection towards that Policy Server is considered failed and torn down. The Net-Net SBC attempts to recreate the TCP connection, followed by the recreating the Diameter connection by issuing a CEA toward the policy server.

Setting the **Keepalive message interval (sec)** parameter to 0 indicates that the Net-Net SBC waits for incoming DWRs, only.

Configuring the Diameter Heart-Beat Mechanism

To configure the Diameter heart-beat mechanism:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click the external policy server you want to configure in the table and click **Edit**.
4. **Keepalive message interval (sec)**—Enter the interval in seconds used to determine a DWA response timeout. The valid range is 0-999999999 seconds. The default value is 0.

Keep alive msg interval (sec)

5. Click **Apply**.

Diameter Failures

During periods of application inactivity on the Diameter interface, Device-Watchdog-Request (DWR) and Answer (DWA) messages are exchanged between the client and server to provide an application-level heartbeat. The exchange frequency of these messages is determined by the Diameter server.

If the Diameter connection fails, the Net-Net SBC tries to re-open the TCP socket and Diameter connection to the Diameter server at 30 second intervals. The Net-Net SBC increases its retry interval to 5 minutes, until a successful Diameter connection is made.

A Diameter connection failure is determined by one of the three events:

1. Diameter Device-Watchdog timeout—The Net-Net SBC detects a timeout when it does not receive a response to the DWR it sent to the Diameter server within the guard timer period. When this happens, the Net-Net SBC tears down the TCP connection and attempts to reconnect to the failed Diameter server.

2. Explicit DPR—The Net-Net SBC detects a DPR message from the Diameter server, and then closes the Diameter connection. Upon receipt of the DPR message, a Disconnect-Peer-Answer (DPA) is returned. The receiver of the DPA initiates the transport disconnection.
3. TCP socket termination—If either side of the Diameter connection receives a FIN or RST, the TCP socket closes per standard behavior. The Net-Net SBC periodically tries to reconnect to the Diameter server.

See the *Diameter Heartbeat* section above for information about configuring Diameter heartbeat.

Application IDs

Diameter messages include an application ID to indicate the application and standards' body interface. The following table lists the different Application-IDs for the corresponding standards' and applications. Application IDs must be provisioned manually.

Standards Reference Point				
RACF			CLF	
AVP	Gq 3GPP R6 29.209	Rx 3GPP R7 29.214	Rq ETSI 283 026	e2 ETSI 283 035
Application-ID	16777222	16777229	16777222	16777231

Diameter Transport-Class/Optional AVP and Application Mode Support

The Net-Net SBC's Diameter interface now supports optional attribute-value pairs (AVPs) for the Resource and Admission Control Function (RACF), and it also now allows you to configure the application mode for the interface.

Optional AVP Support

For both RACF and Connectivity Session Location and Repository Function (CLF) functionality, the Net-Net SBC can now use optional and non-standard AVPs in Diameter messages, including the Transport-Class AVP. Remaining as compliant as possible with current standards, the Net-Net SBC supports non-standard AVPs to extend its base interface modes and thereby increase its operating flexibility.

You can configure one or more AVPs in the external policy server configuration using the options parameter. The Net-Net SBC needs to know in what messages each optional AVP will appear, where the AVP will be located, what AVP flags to set, vendor information, and what data comprises the AVP's payload. Then the Net-Net SBC can insert the optional AVP in outgoing messages and parse incoming messages for it.

About the Transport-Class AVP

When it receives a SIP INVITE triggering external bandwidth management, the Net-Net SBC performs SDP stripping and—through internal processes—selects an external bandwidth manager to use. If the options parameter in the selected external bandwidth manager is set to transport-class, the Net-Net SBC's Diameter RACF interface will issue authentication authorization requests (AARs) with the transport class AVP. The Net-Net SBC does not insert the transport-class AVP messages when the option is not configured.

The transport-class AVP will:

- Be identified with the AVP code 311
- Always have the vendor (V) bit set in the AVP flags
- Never have the mandatory (M) bit set in the AVP flags
- Have the Vendor-Id field set to 13019 (a value specified by ETSI TISPAN)
- Be formatted as an unsigned integer
- Reside in the Media-Component AVP, a grouped AVP

In addition, the transport-class AVP's payload field will be a 32-bit unsigned integer corresponding to a specific media type. The Net-Net SBC learns the specific media type from the m-line of the SDP it received. The following table shows how the Net-Net SBC maps the media types and 32-bit unsigned integers.

m= <media> Incoming SDP flow	Transport Class AVP Value
m=audio	1
m=video	2
m=application	3
m=data	4
m=control	5
m=image	6

New Application Mode Setting

In Release C5.1 and before, the application identifier you set in the external bandwidth manager configuration determines the application mode for the RACF or CLF interface, which would be: Rq, Rx, Gq, and e2. For those releases, you enter an identification number corresponding to the application mode you want to use.

Now, you can also set the application mode to specify the interface more precisely. Doing so avoids the potential for collision between interface types that can occur when you only configure the application identifier. By setting both the application mode and application identifier for the interface, you tell the Net-Net SBC the format for Diameter messages it sends and receives.

The following table describes the application mode settings.

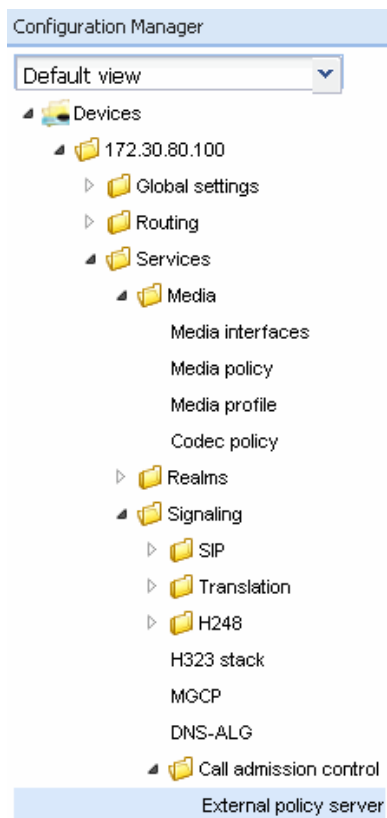
Application Mode Type	Description
Rq	As the default mode for the interface, Rq is the Net-Net SBC's base RACF interface. Even when you leave the application mode set to none (default), the Net-Net SBC runs in Rq mode. The only exception to this rule is if you set the application identifier to 16777236 and leave the application mode set to none; in this instance, the interface runs in Rx mode.
Rx	The interface runs in Rx mode when you either: • Set the application mode to Rx and the application identifier to 16777236 • Leave the application mode set to none, and set the application identifier to 16777236

Application Mode Type	Description
Gq	The interface runs in Gq mode when you set the application mode to Gq. Note that the application identifier 1677722 is no longer unique, but applies both to Rq and Gq interface modes.
e2	The interface runs in e2 mode, the base CLF interface, when you set the application mode to e2. Even if you leave the application mode set to none, the interface will run in e2 mode when the external policy server is configured as a CLF interface.
none	The interface runs in Rq mode when you do not configure an application identifier, or in Rx mode if you set the application identifier to 16777236.

Configuring Optional AVPs

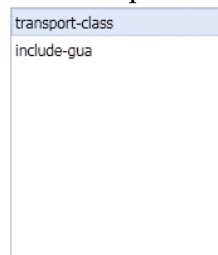
To set the transport-class AVP support for an external bandwidth manager:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



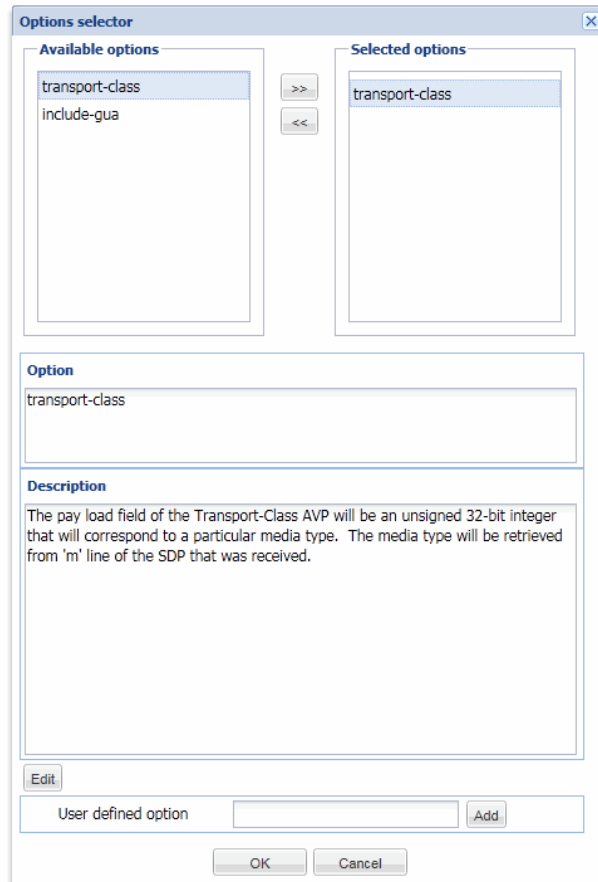
2. Click External policy server.
3. Click an external policy server in the table and click **Edit**. The Edit External policy server window appears.
4. Scroll to Optional AVPs.
5. **Optional AVPs**—Click **Options**. The Options dialog box appears.

6. Click **transport-class** in the Available options list to select it.



transport-class
include-gua

7. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



Options selector

Available options

transport-class
include-gua

Selected options

transport-class

Option

transport-class

Description

The pay load field of the Transport-Class AVP will be an unsigned 32-bit integer that will correspond to a particular media type. The media type will be retrieved from 'm' line of the SDP that was received.

Edit

User defined option Add

OK Cancel

8. Click OK.
9. Click Apply.

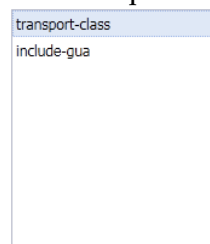
Configuring Service Class Options

To configure service class options:

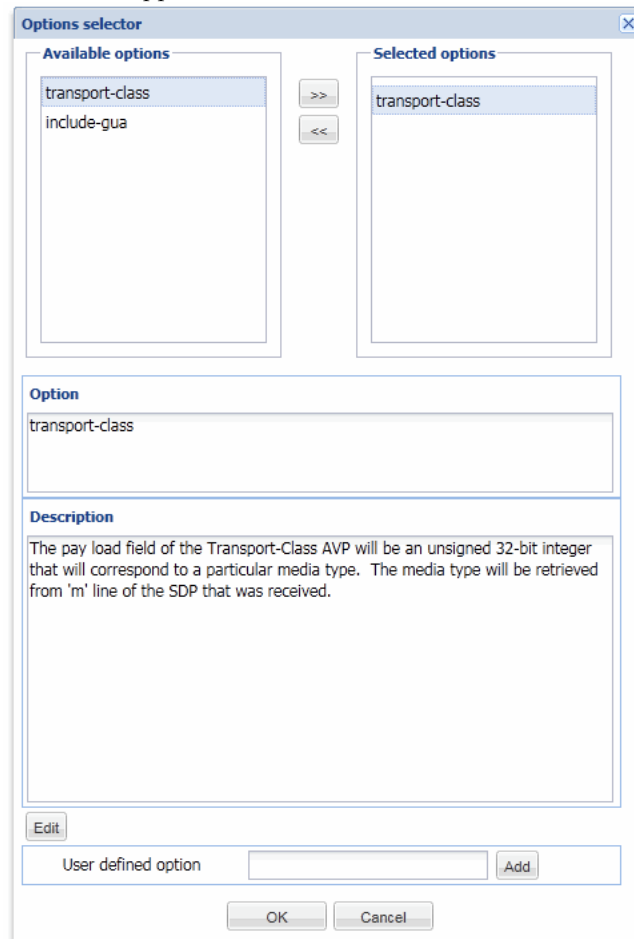
1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click an external policy server in the table and click **Edit**.
4. Scroll to Service class options.
5. **Service class options**—Click **Options**. The Options dialog box appears.
6. Click **transport-class** in the Available options list to select it.



7. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



The dialog box is titled "Options selector" and contains the following sections:

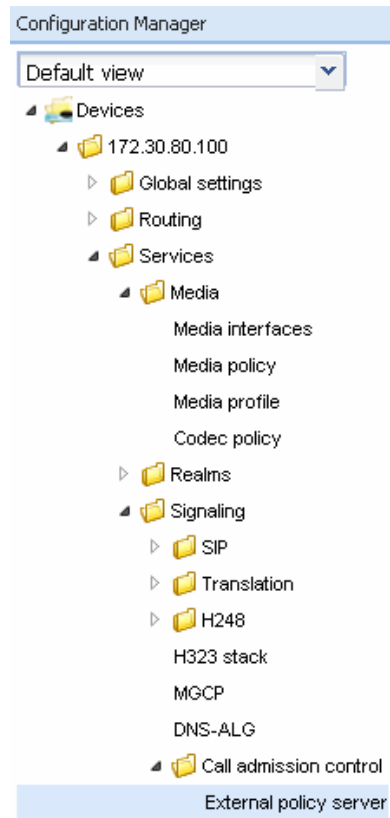
- Available options:** A list box containing "transport-class" and "include-gua".
- Selected options:** A list box containing "transport-class".
- Option:** A text field containing "transport-class".
- Description:** A text area containing the text: "The pay load field of the Transport-Class AVP will be an unsigned 32-bit integer that will correspond to a particular media type. The media type will be retrieved from 'm' line of the SDP that was received."
- Edit:** A button.
- User defined option:** A text field with an "Add" button next to it.
- OK** and **Cancel** buttons at the bottom.

8. Click **OK**
9. Click **Apply**.

Setting the Application Mode

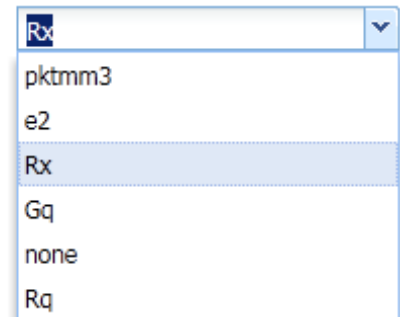
To set the external bandwidth manager's application mode to Rx:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click an external policy server in the table and click **Edit**.
4. **Application mode**—Click **Rx** from the combination box. The default value is none.

Application mode



5. Click **Apply**.

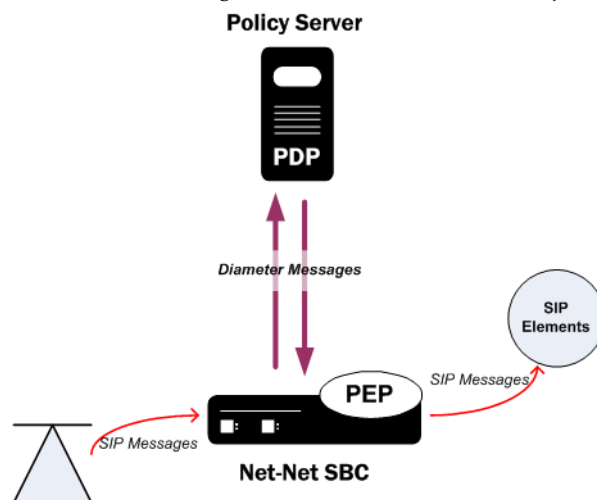
Diameter: Bandwidth-Based Call Admission Control

As the Net-Net SBC proxies and forwards calls, caller media information is known before the INVITE reaches the callee. The Net-Net SBC, acting as a PEP, requests a specific amount of bandwidth for a call, and the RACF can reserve this amount of bandwidth for a call before the called phone rings. A call's required bandwidth can also be reserved by network devices along the path from the caller to callee if the QoS admission criteria is pushed to other PEPs such as routers, along this path to the callee.

Implementation Features

Bandwidth-based CAC is performed according to the following typical scenario. When the Net-Net SBC, known as the Policy Enforcement Point (PEP), receives a SIP INVITE, it sends a Diameter Authentication Authorization Request (AAR) message to the Policy Decision Point (PDP) or Resource and Admission Control Function (RACF). The Net-Net SBC does not forward the INVITE to its destination at this point.

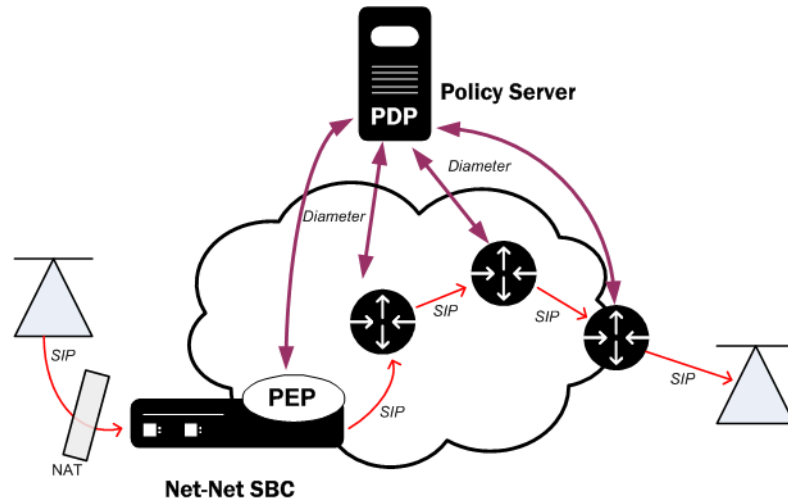
The AAR message includes call identification information and the SDP-based bandwidth requirements for the call. The RACF responds with a Diameter Authentication Authorization Answer (AAA) message to either the install or remove the call. An install command directs the Net-Net SBC to forward the INVITE to the next SIP device. A remove command directs the Net-Net SBC send a SIP 503 Service Unavailable message sent back to the UA and reject the call.



The Net-Net SBC can be configured so that both sides of a call, based on realm, are subject to bandwidth enforcement. Each flow is treated as a unique call/event, because from a media and signaling perspective, they are distinct. As the Net-Net SBC functions as one side of a call, its IP address is inserted into the AAR message regardless of whether it is the calling or called party. This allows for the Diameter install or remove decision to be made before the Net-Net SBC receives the 200 OK response, and before ringing the far-end phone. Only one external policy server can be used within a single realm.

When a call ends, either with the expected SIP BYE or CANCEL methods, or due to other error conditions, the Net-Net SBC alerts the RACF by sending it a Diameter Session Termination Request (STR) message. All ended calls must be deleted from the RACF in order to accurately track used and available bandwidth.

The RACF can apply its hosted policies for calls originating at SIP UAs located behind NATs. This is a standard part of the Net-Net SBC's ability to provide seamless HNT.



Bandwidth Negotiation

Because the decision whether to admit or reject a call is made before the INVITE is forwarded to the called party, some information is not available to the PDP at the initial request. The final IP Address, UDP port number, that transport the RTP flow, and the codec used are not known by the Net-Net SBC until the called party responds with its own SDP information (either in the 180 or 200 response).

The Net-Net SBC examines the Session Description Protocol (SDP) value in the body of the SIP INVITE to determine what codecs are available for the call. If the INVITE specifies more than one codec, the Net-Net SBC bases its request to the RACF on the most bandwidth-hungry codec to ensure that all bandwidth requests will succeed or fail on the first attempt.

If the call is admitted, and when the called party returns finalized SDP information, the Net-Net SBC modifies the original reservation with the chosen codec's bandwidth requirements. This ensures the RACF has current and accurate information with which to make policy decisions.

Session Lifetime

When receiving a successful Diameter response message for bandwidth from the RACF, a session lifetime timer is included in the message. This timer states how long the session can last. If the session continues past 3/4 of session lifetime, the Net-Net SBC sends another bandwidth request for that session to ultimately refresh the lifetime timer. If the RACF grants this bandwidth request, the Net-Net SBC continues to allow the session to proceed uninterrupted. If a lifetime timer for a session is not returned to the Net-Net SBC by the RACF, the Net-Net SBC assumes the session can last forever and never issues a refresh in this manner.

Net-Net High Availability Support for CAC

The Net-Net SBC's high availability (HA) capabilities support CAC. When one Net-Net SBC in an HA configuration goes out of service, the MAC addresses are reassigned to a healthy Net-Net SBC. IP addresses "follow" the MAC addresses to provide a seamless switchover between HA nodes.

After an HA failover, the Diameter connection on the primary Net-Net SBC is either gracefully torn down, or times out depending on behavior of the PDP. The backup Net-Net SBC attempts to create a new Diameter connection with the PDP.

Diameter AAR Query Post SDP Exchange

For Diameter, the Net-Net SBC supports sending the Authentication-Authorize-Request (AAR) query upon SDP answer instead of the SDP offer. This change can be useful in WiMax environments where mobile phones go idle and become semi-detached from their base stations and from the WiMax access controller (WAC). In such a case, the WAC receives an AAR from the idle user but, because it cannot determine that user's base station, rejects the request.

The Net-Net SBC's external policy server support offers three ways for you to handle the reservation of bandwidth for incomplete flows. You set these in the **Early resource reservation** parameter.

- **Enabled:** supports the usual behavior when the AAR is sent upon SDP offer as well as SDP answer. This mode ensures backwards compatibility.
- **Orig-realm-only:** allows calls originating from a realm with a policy server associated with it to send the AAR upon SDP offer. However, calls terminating at a realm with a policy server associated with it send the AAR post SDP exchange.
- **Disabled:** allows no bandwidth reservation for incomplete flows.

Configuring AAR Post SDP Exchange

You can configure this behavior using the **Early resource reservation** parameter in the **External policy server** configuration. This feature works for COPS as well as Diameter.

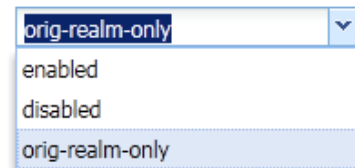
To send the AAR post SDP exchange:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click an external policy server in the table and click **Edit**.
4. **Early resource reservation**—Click **orig-realm-only** from the combination box to choose it to ensure that calls terminating at realms with policy servers send their AARs after the SDP exchange. The values for this parameter are:
 - **Enabled** (default): supports the usual behavior when the AAR is sent upon SDP offer as well as SDP answer. This mode ensures backwards compatibility.
 - **Orig-realm-only**: allows calls originating from a realm with a policy server associated with it to send the AAR upon SDP offer. However, calls terminating at a realm with a policy server associated with it send the AAR post SDP exchange.
 - **Disabled**: allows no bandwidth reservation for incomplete flows.

Early resource reservation



5. Click **Apply**.

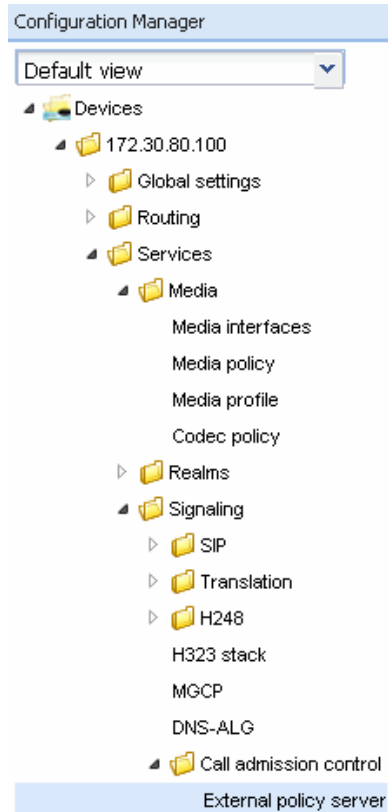
Diameter: Configuring RACF

In the following configuration procedures, we assume that your baseline configuration passes SIP traffic; with the Net-Net SBC in the role of an access SBC.

Configuring the External Policy Server

To configure the external policy server:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



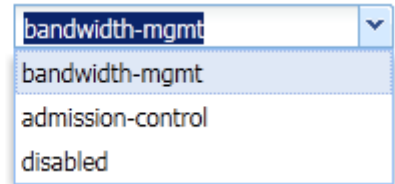
2. Click External policy server.
3. Click **Add** to create a new external policy server. The Add External policy server dialog box appears.
4. **Name**—Enter the name of the external policy server.
5. Click **Apply**.

 A screenshot of the 'Add External policy server' dialog box. It has a title bar that says 'Add External policy server'. Inside, there is a label '*Name' followed by a text input field containing 'ExtPolicyServer1'. To the right of the input field is the text '(Unique required)'. At the bottom right, there are two buttons: 'Apply' and 'Cancel'.

6. Select the new external policy server in the table and click **Edit**.
7. **State**—Select enabled from the combination box to enable this server.

8. **Operation mode**—Choose bandwidth-mgmt from the combination box to enable bandwidth management.

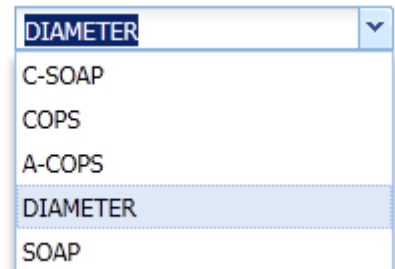
Operation mode



A dropdown menu with a blue border. The selected item is 'bandwidth-mgmt' in a blue box. Below it, the list of options is shown: 'bandwidth-mgmt', 'admission-control', and 'disabled'.

9. **Policy communication protocol**—Choose DIAMETER from the combination box to support Diameter-based CAC.

Policy communication protocol



A dropdown menu with a blue border. The selected item is 'DIAMETER' in a blue box. Below it, the list of options is shown: 'C-SOAP', 'COPS', 'A-COPS', 'DIAMETER', and 'SOAP'.

10. **IP address**—Enter the IP address of the external PDP.
11. **Port number**—Enter the port number the diameter connection connects to on the PDP.

IP address

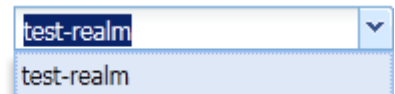
192.168.40.30

Port number

80

12. **Realm in which server resides**—Choose the name of the realm in which this Net-Net SBC defines the PDP to exist from the combination box. This is not necessarily the realm where the Net-Net SBC performs admission control.

Realm in which server resides



A dropdown menu with a blue border. The selected item is 'test-realm' in a blue box. Below it, the list of options is shown: 'test-realm'.

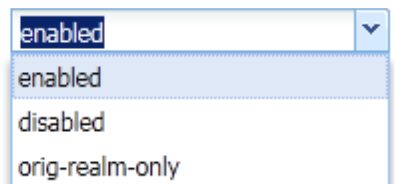
13. **Number of policy connections**—Enter the number of policy protocol TCP connections to establish to the PDP. For Diameter, this should be the value 1.

Number of policy connections

1

14. **Early resource reservation**—Select enabled from the combination box to enable when communicating with a PDP via Diameter. The Net-Net SBC makes admission requests before learning all the details of the flows and devices (e.g., not knowing the final UDP port numbers for the RTP media streams until after the RTP has begun).

Early resource reservation



A dropdown menu with a blue border. The selected item is 'enabled' in a blue box. Below it, the list of options is shown: 'enabled', 'disabled', and 'orig-realm-only'.

15. Click **OK**. The new external policy server name appears in the list.

16. Click the new external policy name in the table and click **Edit**. The Edit External policy sever dialog box appears.
17. **Permit call when connection down**—Select enabled from the combination box to enable the external policy server configuration to permit connection down.

Permit call when connection down

enabled
disabled

18. **Product name**—Enter text string that describes the vendor-assigned name for the RACF. This parameter is required.
19. **Application id**—Enter a numeric application ID that describes the interface used to communicate with the RACF.

Product name

Application ID

20. **Framed ip address encode**—Choose the format of the Frame-IP-Address (AVP-8) in Diameter messages.

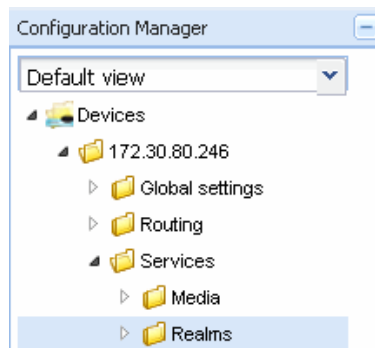
Framed IP address encode

octet-string
ascii-string

Configuring Realms for Diameter Support

To configure the realm for Diameter support:

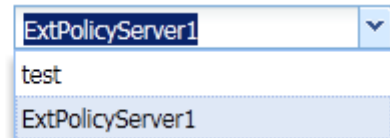
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the Realm table and click **Edit**.
4. Scroll to Resource Admission Control Function (RACF).

5. **External policy server**—Choose the name of the external policy server to be used for external CAC for this Realm from the combination box.

External policy server



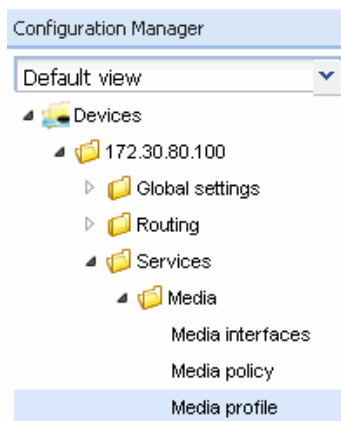
6. Click **Apply**.

Configuring Media Profiles for CAC

Values for the following parameters can be found in the PacketCable™ Audio/Video Codecs Specification PKT-SP-CODEC-I06-050812 document.

To configure the media profile for Diameter support in CAC:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click the media profile you want to configure in the table and click **Edit**.
4. **Peak rate limit (bytes/sec)**—Enter the r, P (Token bucket rate / Peak data rate) value.
5. **Maximum burst size (bytes)**—Enter the b, m, M (Token bucket size / Min policed unit) value.

Peak rate limit (bytes per second)

55

Maximum burst size (bytes)

66

6. Click **Apply**.

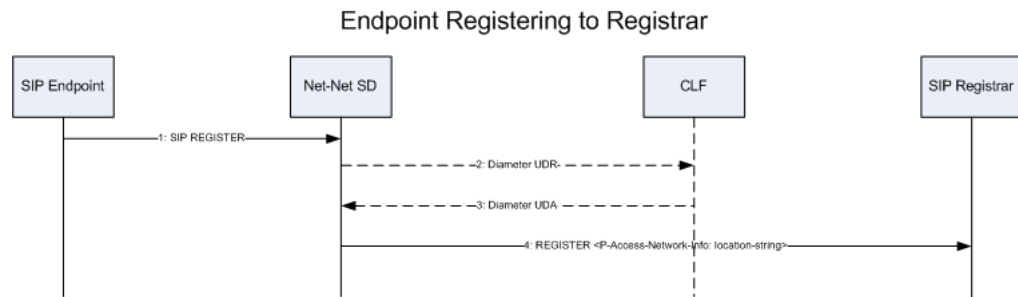
Diameter: Connectivity Location Function

A Connectivity Location Function (CLF) maintains mappings between endpoints with dynamically assigned IP addresses and their physical location. The Net-Net SBC, acting as a P-CSCF, is the intermediary device between a registering endpoint and a CLF. The CLF thus validates and tags a registering endpoint, and the Net-Net SBC applies the CLF's actions. The Net-Net SBC supports both COPS and Diameter protocols to maintain a connection with the CLF.

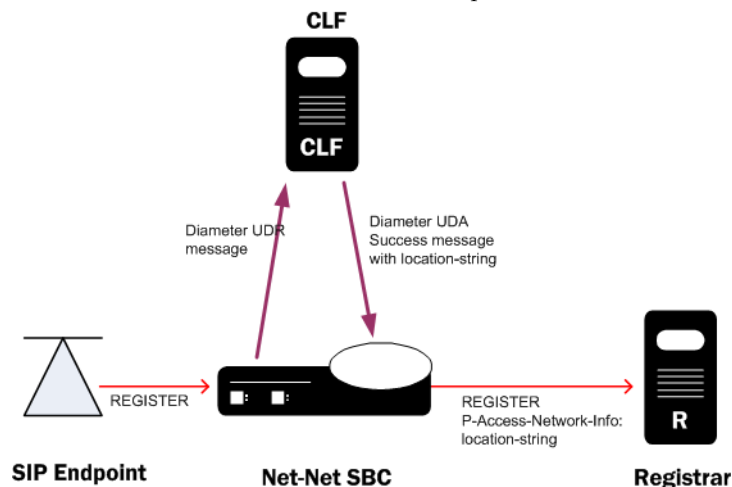
CLF Behavior

The Net-Net SBC and a CLF only interact with each other when an endpoint registers or re-registers. The Net-Net SBC, acting as the P-CSCF, is the first SIP device that the REGISTER message reaches. Upon receiving the REGISTER message(1), the Net-Net SBC queries the CLF using the Diameter protocol. The endpoint's (public) IP address and port, and the Net-Net SBC's IP information are sent to the CLF in a Diameter User-Data-Request (UDR) message(2).

The CLF responds to the Net-Net SBC with a Diameter User-Data-Answer (UDA) message(3). If the request is approved, then the CLF also sends a location-string value to be inserted in one of the SIP headers. The Net-Net SBC inserts a P-Access-Network-Info header containing the location-string into the incoming REGISTER message and forwards this message(4) to the SIP registrar/I/S-CSCF.



The Net-Net SBC inserts this P-Access-Network-Info header into all subsequent SIP messages from this endpoint as they are forwarded into the core network. The P-Access-Network-Info header is inserted into all SIP requests and responses except for ACK and CANCEL messages. For all boundaries where SIP messages pass from trusted to untrusted SIP interfaces or session agents, the Net-Net SBC will strip out the P-Access-Network-Info header as expected.



If the CLF responds with a Reject UDA message, the Net-Net SBC rejects the registration, and sends a 503 - Service Unavailable message back to the registering endpoint. In this way, the CLF can be used for admission control.

The Net-Net SBC communicates with the CLF solely for retrieving location information from the CLF, and not for notifying the CLF about an endpoint's registration state or activity. When an endpoint's registration ends, either through a normal expiration, getting rejected by the registrar, or through specific de-registering or error conditions, the Net-Net SBC deletes the locally cached

registration location string. The Net-Net SBC does not inform the CLF about any registrations that have been deleted.

P-Access-Network-Info Header Handling

The P-Access-Network-Info header is created and populated according to the following rules:

1. If the CLF returns an Accept UDA message with a location string, the Net-Net SBC inserts the location string into a P-Access-Network-Info header in the outgoing REGISTER message.
2. If the CLF returns an Accept UDA message without a location string, the Net-Net SBC inserts the configured default string into a P-Access-Network-Info header in the outgoing REGISTER message.
3. If the CLF returns an Accept UDA message without a location string and no location string is configured on Net-Net SBC, the outgoing REGISTER message is forwarded out of the Net-Net SBC, but no P-Access-Network-Info header is created for the REGISTER message.

CLF Re-registration

The Net-Net SBC will send a new UDR message to the CLF to request a new location string if any of the following events occur:

1. The endpoint's contact address changes.
2. The SIP Register message's Call-ID header changes.
3. The endpoint's public IP Address or UDP port changes.
4. The endpoint connects to a different SIP interface, port, or realm on the Net-Net SBC than it did in the initial REGISTER message.
5. The registration expires in the Net-Net SBC's registration cache.

CLF Failures

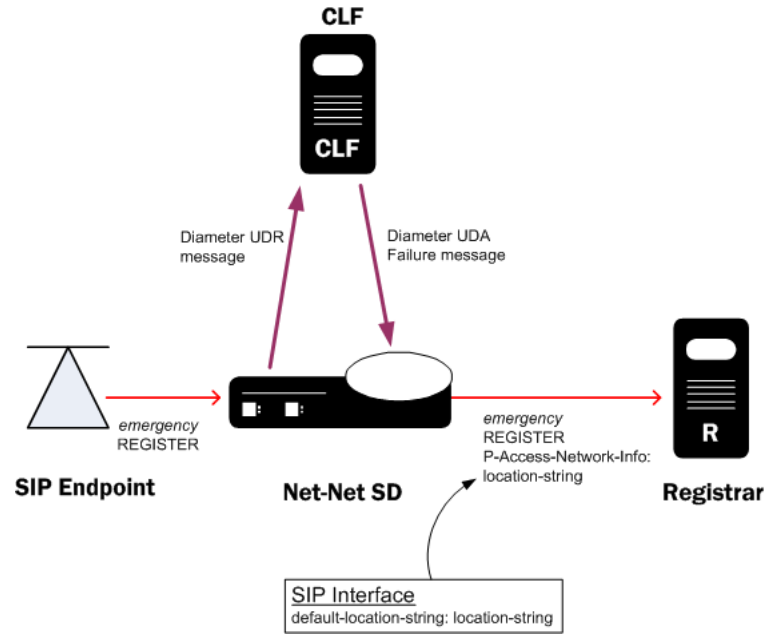
If a Diameter connection fails, the Net-Net SBC will continually try to re-establish the connection. Endpoints that are already registered will stay registered unless they timeout or if the registrar rejects their refreshes. When the Diameter connection has not been established, and an endpoint registers on a SIP interface that is configured to use CLF, the Net-Net SBC forwards new REGISTER messages to the registrar using the default location string.

CLF Emergency Call Handling

The Net-Net SBC allows emergency calls into the network even if the endpoint that places the emergency call is not registered. In the expected fashion, the Net-Net SBC will query the CLF first for an incoming emergency call sourced from an unregistered endpoint. If the CLF response is successful, then the Net-Net SBC will insert the string returned from the CLF into a P-Access-Network-Info header, and insert this header into the emergency call's REGISTER message. If no location string is returned with a successful CLF response, the default location string is inserted into P-Access-Network-Info header.

If the CLF's response is to reject the emergency call, the Net-Net SBC will insert the configured default location string into the P-Access-Network-Info header and forward the emergency call's REGISTER message toward the registrar. For

emergency calls where the endpoint has already successfully registered, the call will be routed into the network using the expected methods for emergency call routing.



If the Diameter connection to the CLF is down, emergency calls from un-registered endpoints are still allowed into the network using the default string inserted into the emergency messages.

HA Functionality

The location strings generated by the CLF are replicated on the standby SD in an HA pair. This is required so that a Net-Net SBC in an HA pair can instantly continue processing calls using the previously learned CLF information.

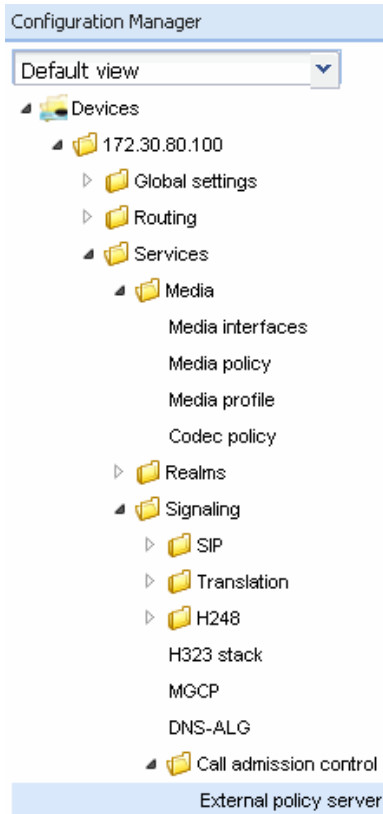
Diameter: Configuring CLF

In the following configuration examples, we assume that your baseline configuration passes SIP traffic, with the Net-Net SBC in the role of an Access SBC. In this example, you will configure additions to the realm configuration and the new external policy server configuration.

Configuring External Policy Server for CLF

To configure the external policy server for use with a CLF:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



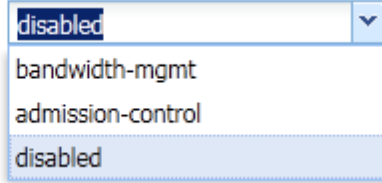
2. Click External policy server.
3. Click **Add** to create a new external policy server. The Add External policy server dialog box appears.
4. **Name**—Enter the name of the external policy server.
5. Click **Apply**.

The screenshot shows the 'Add External policy server' dialog box. It has a text field for '*Name' with the value 'ExtPolicyServer1' entered. To the right of the text field is the text '(Unique required)'. At the bottom of the dialog box are two buttons: 'Apply' and 'Cancel'.

6. Select the new external policy server in the table and click **Edit**.
7. **State**—Select enabled from the combination box to enable this server.

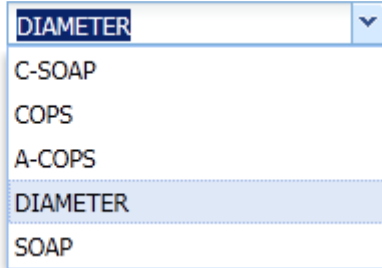
8. **Operation mode**—Choose **admission-control** from the combination box to let the Net-Net SBC communicate with a CLF.

Operation mode



9. **Policy communication protocol**—Choose **Diameter** from the combination box to connect with a CLF.

Policy communication protocol



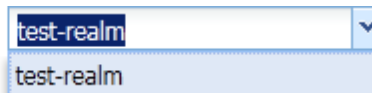
10. **IP address**—Enter the IP address of the CLF.
11. **Port number**—Enter the port number the diameter that the CLF uses for Diameter transactions.

IP address

Port number

12. **Realm in which server resides**—Choose the name of the realm where the CLF exists from the combination box. This is not necessarily the realm where the Net-Net SBC performs admission control.

Realm in which server resides

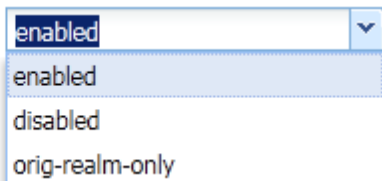


13. **Number of policy connections**—Enter the number of connections the Net-Net SBC will create with the CLF. For Diameter, this should be the value 1.

Number of policy connections

14. **Early resource reservation**—Select enabled from the combination box to enable the Net-Net SBC to send a message to the CLF that does not include the endpoint's true port number. A value of 0 will be used for the port number.

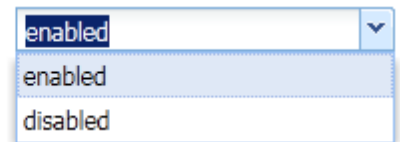
Early resource reservation



15. Click **OK**. The new external policy server name appears in the list.
16. Click the new external policy name in the table and click **Edit**. The Edit External policy sever dialog box appears.

17. **Permit call when connection down**—Select enabled from the combination box to enable the external policy server configuration to permit connection down.

Permit call when connection down



18. **Product name**—Enter a text string that describes the vendor-assigned name for the CLF. This parameter is required.

Product name

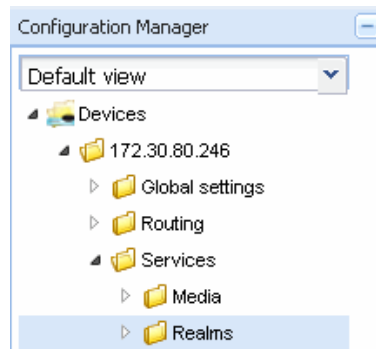
CLF Application

19. Click **Apply**.

Configuring the SIP Interface for CLF

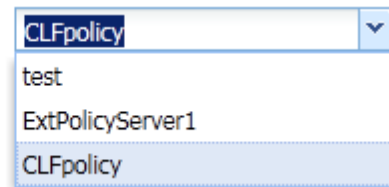
To configure the SIP interface for CLF support:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Click the realm you want to configure in the table and click **Edit**.
4. Click the Service/Protocols tab. The Realm services table appears.
5. Click the realm you want to configure in the table and click **Edit**.
6. With the General tab selected, scroll to Connectivity Session Location Function (CLF).
7. **Policy server**—Choose the name of the external policy server you used for the CLF server from the combination box.

Policy server



8. **Default-location-string**—Enter the default location string you want inserted into a P-Access-Network-Info header when the CLF server does not return a unique location string.

Default location string

clflocation

9. Click **Apply**.

Diameter e2

The Net-Net SBC accepts, parses, and processes the Experimental-Result AVP (which indicates that an error has occurred) whether accompanied by the Result-Code AVP or not. When a CLF or RACF interface receives the Experimental-Result AVP, it maps returned and corresponding values to construct responses for the requesting signaling application (SIP).

How It Works: CLF

This section explains how the Net-Net SBC's Diameter CLF interface handles the Experimental-Results AVP and the Result-Code AVP.

CLF Experimental Result Handling

When the Diameter CLF interface receives a User-Data-Answer (UDA) message that contains the Experimental-Result AVP, it parses that AVP. The Experimental-Result AVP is a grouped AVP that includes the Experimental-Result-Code AVP, which contains an enumerated value in its payload that specifies the status of the received UDA message—either success or failure. In the case where the enumerated value indicates:

- Success—The Diameter CLF interface internally notifies SIP signaling of its status.
- Failure (or non-success)—The Diameter CLF interface communicates internally with the Net-Net SBC's SIP signaling application to instruct it to use the configured default string in the P-Access-Network-Info header when the value received is either: `DIAMETER_ERROR_USER_UNKNOWN` or `DIAMETER_USER_DATA_NOT_AVAILABLE`.

The Net-Net SBC forwards the Register using the default location string (or without the PANI header if no location string is configured) when the value received is `DIAMETER_UNABLE_TO_COMPLY`.

For all other such result codes, the Register is rejected.

CLF Result Code Handling

When the Diameter CLF interface receives a User-Data-Answer (UDA) message that contains the Result-Code AVP, it determines and performs actions based on the AVP's enumerated value that indicates success or failure:

- Success—In the case where the enumerated value indicates success, the Net-Net SBC's Diameter CLF interface internally notifies SIP signaling of its status.
- Failure—In the case where the enumerated value indicates failure (or non-success), the Diameter CLF interface communicates internally with the Net-Net SBC's SIP signaling application to instruct it to use the configured default string in the P-Access-Network-Info header when the value received is either: `DIAMETER_UNABLE_TO_COMPLY`. Other failure values will be treated as absolute, and requests will be rejected on that basis.

How It Works: RACF Experimental Result Handling

This section explains how the Net-Net SBC's Diameter RACF interface handles the Experimental-Results AVP.

When the Diameter RACF interface receives an Authentication-Authorization-Answer (AAA) message that contains the Experimental-Result AVP, it will parse that AVP. The Experimental-Result AVP is a grouped AVP that includes the Experimental-Result-Code AVP, which contains an enumerated value in its payload that specifies the status of the received AAA message—either success or failure. In the case where the enumerated value indicates:

- Success—The Diameter RACF internally notifies SIP signaling of its status.
- Failure (or non-success)—The Diameter RACF interface communicates internally with the Net-Net SBC's SIP signaling application, which will then reject the request with a 503 Service Unavailable response.

About Realms and e2 Enhancements

This section describes how the Net-Net SBC treats payload format strings for destination, origin, and host realms.

Destination Realms

The Diameter CLF and RACF interfaces can change the format of the payload string in the Destination-Realm AVP for any Diameter message it originates and sends to an external server. The payload field for this AVP can be constructed in any the following formats:

Format	Description
user-with-realm	• user—IP address of the endpoint initiating the call with the Net-Net SBC • realm—Name of the realm on which the Net-Net SBC received the INVITE from a user
user-only	• user—IP address of the endpoint initiating the call with the Net-Net SBC
realm-only	• realm—Name of the realm on which the Net-Net SBC received the INVITE from a user

When either the Diameter CLF or RACF interface sends any message with the Destination-Realm AVP, it determines from the external policy server configuration how to construct the payload string for this AVP.

You can set the format to use in the external policy server configuration. You can use any value from the table above; the default is `<user>@<realm>`. By treating the format this way, the policy server and the Net-Net SBC can easily communicate this value; if sent to the policy server in any AVP, the policy server can simply return the full value.

Origination and Host Realms

The Diameter CLF and RACF interfaces can change the suffix for Origin-Realm and Origin-Host AVPs that have a payload string constructed as a domain name.

You can set the suffix you want appended to payload strings in the external policy server configuration. This parameter can be set to any string (default is `.com`), and the Net-Net SBC automatically adds a dot (.) to the front of this entry if you do not include one. The policy server and the Net-Net SBC can easily communicate this value; if sent to the policy server in any AVP, the policy server can simply return the full value.

Configuring Diameter e2

This section shows you how to set the format to use for Destination-Realm AVPS and how to configure a domain name suffix for Origin-Realm and Origin-Host AVPs.

To set the format to use for the payload string in the Destination-Realm AVP:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click the External policy server tab. The External policy table appears.
4. Click the external policy you want to edit in the table and click **Edit**.
5. Scroll to Destination realm AVP.
6. **Format**—Choose the format you want to use for the Destination-Realm AVP from the combination box.

Format user-with-realm

The following table lists the format options.

Format	Description
user-with-realm	• user—IP address of the endpoint initiating the call with the Net-Net SBC • realm—Name of the realm on which the Net-Net SBC received the INVITE from a user
user-only	• user—IP address of the endpoint initiating the call with the Net-Net SBC
realm-only	• realm—Name of the realm on which the Net-Net SBC received the INVITE from a user

7. **Domain name suffix**—Enter the suffix you want to use for Origin-Realm and Origin-Host AVPs that have a payload string constructed as a domain name

Your value can be any string, to which the Net-Net SBC will prepend a dot if you do not include one.

Domain name suffix

.com

8. Click **Apply**.

Diameter Gq Proxy Interface and Bit Enhancements

The Net-Net SBC supports the Gq interface for the External policy server Diameter protocol.

Proxy Bit Enhancements

When a signaling protocol receives an event request, the Net-Net SBC must ensure the external policy server on the other end has enough bandwidth to maintain the requested call. The SDP information from the signaling message is stripped and encoded into the Diameter Band Request to be forwarded onto the external policy server.

In previous releases, this external policy server either handled the Diameter message on its own, or sent it to a proxy server, depending on how much traffic it was handling at the time. This was done without any input from the Net-Net SBC.

The proxy bit allows the Net-Net SBC to tell the external policy server whether it wants the main server to handle the Diameter message, or if it can proxy it to another server on the network.

Configuring Diameter Gq

When the `Set proxy server flag` parameter is enabled, the proxy bit is set and the external policy server must process this Diameter request. When the parameter is disabled, the Net-Net SBC gives the external policy server permission to proxy the request along.

Configuring External Policy Servers

To configure Diameter Gq for the external policy server:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click the external policy server name in the table and click **Edit**.
4. **Application id**—Enter the numeric application ID 16777222, which specifies this interface will be used in Gq mode.

Application ID

16777222

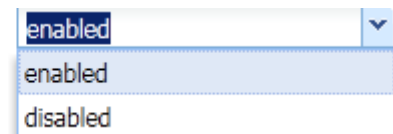
5. Click **Apply**.

Enabling the Proxy Bit

To enable the proxy bit in the Diameter header:

1. From the External policy server table, click the external policy server name in the table
2. Click **Edit**.
3. **Set proxy server flag**—Select enabled from the combination box to enable the proxy bit in the header.

Set proxy server flag



4. Click **Apply**.

Diameter Rx Enhancements

You can configure the Diameter interface on your Net-Net SBC to run in Rx mode, which brings the Net-Net SBC's Diameter interface in line with the 3GPP Rx reference point specifications.

The Net-Net SBC Diameter interface's operation is tied to events for SIP signaling that trigger external bandwidth management. Currently, SIP is the only supported signaling protocol for Diameter external bandwidth management.

How it Works

You can configure the Net-Net SBC to run in Rx mode by setting the external proxy servers application ID to 16777236 (the Rx application identifier). This value identifies this interface as the Rx interface. For all Authentication-Authorization-Request messages the Net-Net SBC sends over its Diameter interface:

- The Reservation-Priority AVP (Attribute Value Pair) is included for priority calls. This AVP will be the main AVP within the AAR message.
- The Codec-Data AVP is included for non-priority calls. This AVP is one of several that together comprise a Group AVP structure.

Non-Priority Call Handling

When a SIP signaling event triggers external bandwidth management use, the Net-Net SBC removes all SDP information from the signaling message that was the trigger. The Net-Net SBC repackages this bandwidth information so that it can form a Bandwidth Request and decides on an external bandwidth manager to which it should be sent. If the appropriate external bandwidth manager is configured for Rx interface use, then Net-Net SBC reformats the SDP information to construct a Codec-Data AVP.

If the external bandwidth manager receiving the request ignores the SDP information, it will not include the Codec-Data AVP in the AAR.

For calls that do not require special treatment, the Codec-Data AVP is required to have the:

- AVP code 524
- 3GPP vendor identification number (10415)
- "V" (Vendor) bit set in the AVP
- "M" (Mandatory) bit set when sending this AVP
- Type octet string

In addition, the Codec-Data AVP must be encoded as described in the following table.

AVP section/line	Requirement
Line 1	Must specify the direction of the flow by including the ASCII “uplink” or “downlink”: • uplink—Identifies that the SDP came from the UE and was sent to the network • downlink—Identifies that the SDP came from the network and was sent to the UE
Line 2	Must specify whether the offer or answer codec is at issue by including the ASCII “offer” (from an SDP offer according to RFC 3264) or “answer” (from an answer according to RFC 3264)
Remainder of the AVP	Must include lines found in the signaling SDP, formatted in ASCII and separated by new-line characters; the first line of this section must be the “m” line, followed by any “a” or “b” lines related to that “m” line

Priority Call Handling

The Net-Net SBC determines a call is a priority call when it matches a defined network management control (NMC) priority rule. No other scenario triggers the priority call handling treatment.

When a SIP signaling event triggers external bandwidth management use for a priority call, the Net-Net SBC forms the Band Request reflecting the call’s priority status and determines which external bandwidth manager to use. If the appropriate external bandwidth manager is configured for Rx interface use, then Net-Net SBC reformats the SDP information to construct a Reservation-Priority AVP and includes it in the AAR message.

For priority calls, the Reservation-Priority AVP is included in the AAR and is required to:

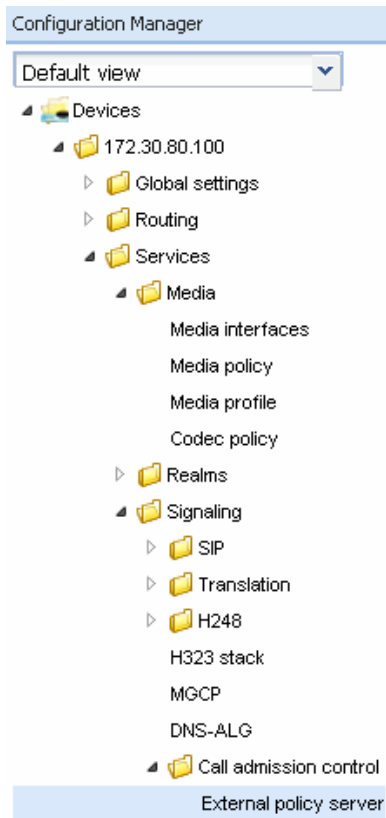
- Use the ETSI Vendor identification number (13019)
- Have the “V” (Vendor) bit set in the AVP
- Not to have the “M” (Mandatory) bit set when sending this AVP
- Be of the type enumeration with the following possible values: 0—default or 1 through 7—Priorities one through seven, where the level of priority increases with numeric value

For the Net-Net SBC, the Reservation-Priority AVP will be set to PRIORITY-SEVEN (7) if it is present.

Configuring Rx Mode

To configure Rx mode:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click the external policy server name in the table and click **Edit**. The Edit External policy server window appears.
4. **Application id**—Enter the numeric application ID 16777236, which specifies this interface will be used in Rx mode.

Application ID

16777222

5. Click **Apply**.

Diameter Transport-Class/Optional AVP and Application Mode Support

The Net-Net SBC's Diameter interface now supports optional attribute-value pairs (AVPs) for the Resource and Admission Control Function (RACF), and it also now allows you to configure the application mode for the interface.

Optional AVP Support

For both RACF and Connectivity Session Location and Repository Function (CLF) functionality, the Net-Net SBC can now use optional and non-standard AVPs in Diameter messages, including the Transport-Class AVP. Remaining as compliant as possible with current standards, the Net-Net SBC supports non-standard AVPs to extend its base interface modes and thereby increase its operating flexibility.

You can configure one or more AVPs in the external policy server configuration using the options parameter. The Net-Net SBC needs to know in what messages each optional AVP will appear, where the AVP will be located, what AVP flags to set, vendor information, and what data comprises the AVP's payload. Then the Net-Net SBC can insert the optional AVP in outgoing messages and parse incoming messages for it.

About the Transport-Class AVP

When it receives a SIP INVITE triggering external bandwidth management, the Net-Net SBC performs SDP stripping and—through internal processes—selects an external bandwidth manager to use. If the options parameter in the selected external bandwidth manager is set to transport-class, the Net-Net SBC's Diameter RACF interface will issue authentication authorization requests (AARs) with the transport class AVP. The Net-Net SBC does not insert the transport-class AVP messages when the option is not configured.

The transport-class AVP will:

- Be identified with the AVP code 311
- Always have the vendor (V) bit set in the AVP flags
- Never have the mandatory (M) bit set in the AVP flags
- Have the Vendor-Id field set to 13019 (a value specified by ETSI TISPAN)
- Be formatted as an unsigned integer
- Reside in the Media-Component AVP, a grouped AVP

In addition, the transport-class AVP's payload field will be a 32-bit unsigned integer corresponding to a specific media type. The Net-Net SBC learns the specific media type from the m-line of the SDP it received. The following table shows how the Net-Net SBC maps the media types and 32-bit unsigned integers.

m= <media> Incoming SDP flow	Transport Class AVP Value
m=audio	1
m=video	2
m=application	3
m=data	4

m= <media> Incoming SDP flow	Transport Class AVP Value
m=control	5
m=image	6

New Application Mode Setting

In Release C5.1 and before, the application identifier you set in the external bandwidth manager configuration determines the application mode for the RACF or CLF interface, which would be: Rq, Rx, Gq, and e2. For those releases, you enter an identification number corresponding to the application mode you want to use.

Now, you can also set the application mode to specify the interface more precisely. Doing so avoids the potential for collision between interface types that can occur when you only configure the application identifier. By setting both the application mode and application identifier for the interface, you tell the Net-Net SBC the format for Diameter messages it sends and receives.

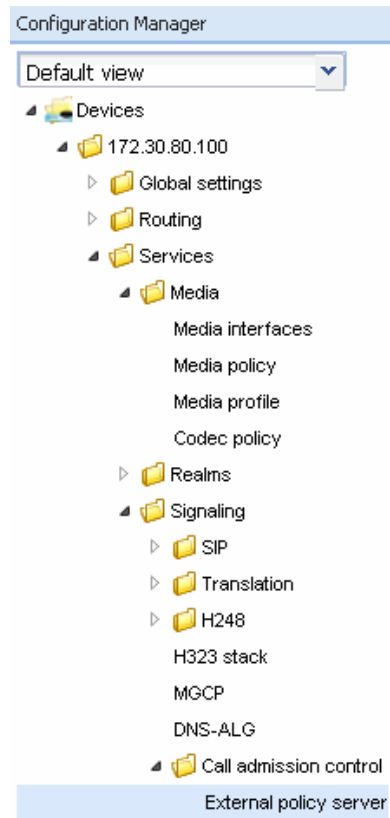
The following table describes the application mode settings.

Application Mode Type	Description
Rq	As the default mode for the interface, Rq is the Net-Net SBC's base RACF interface. Even when you leave the application mode set to none (default), the Net-Net SBC runs in Rq mode. The only exception to this rule is if you set the application identifier to 16777236 and leave the application mode set to none; in this instance, the interface runs in Rx mode.
Rx	The interface runs in Rx mode when you either: - Set the application mode to Rx and the application identifier to 16777236 - Leave the application mode set to none, and set the application identifier to 16777236
Gq	The interface runs in Gq mode when you set the application mode to Gq. Note that the application identifier 1677722 is no longer unique, but applies both to Rq and Gq interface modes.
e2	The interface runs in e2 mode, the base CLF interface, when you set the application mode to e2. Even if you leave the application mode set to none, the interface will run in e2 mode when the external policy server is configured as a CLF interface.
none	The interface runs in Rq mode when you do not configure an application identifier, or in Rx mode if you set the application identifier to 16777236.

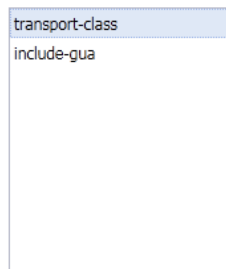
Configuring Optional AVPs

To set the transport-class AVP support for an external bandwidth manager:

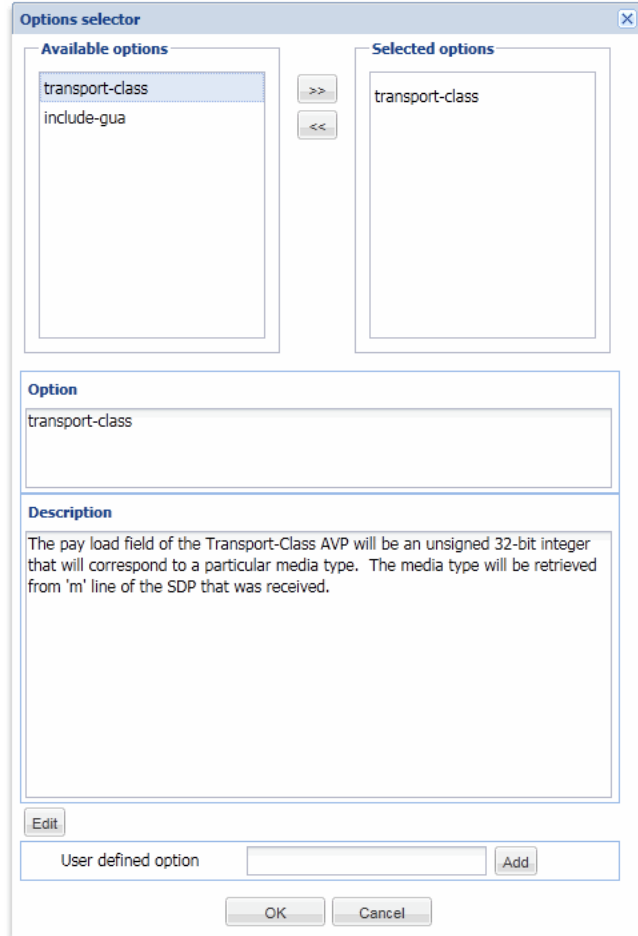
1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click an external policy server in the table and click **Edit**.
4. Click **Options**. The Options dialog box appears.
5. Click **transport-class** in the Available options list to select it.



6. Click  to move the option to the Selected options list. The option name and details appear at the bottom of the screen.



The dialog box is titled "Options selector" and contains the following elements:

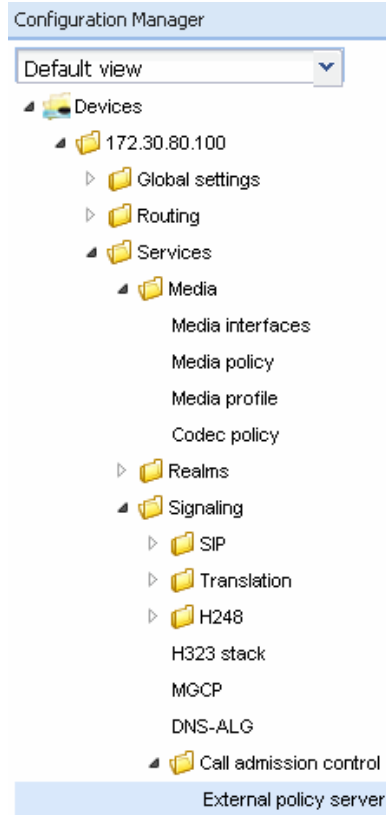
- Available options:** A list box containing "transport-class" and "include-gua".
- Selected options:** A list box containing "transport-class".
- Navigation buttons:** ">>" and "<<" buttons between the two lists.
- Option field:** A text box containing "transport-class".
- Description field:** A text area containing the text: "The pay load field of the Transport-Class AVP will be an unsigned 32-bit integer that will correspond to a particular media type. The media type will be retrieved from 'm' line of the SDP that was received."
- Edit button:** A button labeled "Edit" below the description field.
- User defined option:** A text box with the label "User defined option" and an "Add" button.
- OK and Cancel buttons:** Buttons at the bottom of the dialog.

7. Click OK
8. Click Apply.

Setting the Application Mode

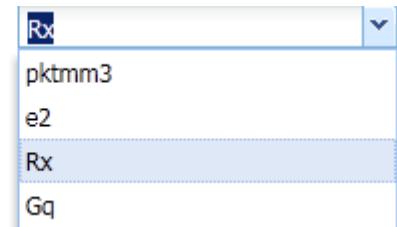
To set the external bandwidth manager's application mode to Rx:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click an external policy server in the table and click **Edit**.
4. **Application mode**—Click Rx from the combination box. The default value is none.

Application mode



5. Click **Apply**.

Diameter Media Policing Enhancements

When you use the Net-Net SBC's Diameter support for bandwidth management, you can now configure:

- The exact amount of bandwidth being policed on the Net-Net SBC internally, and being policed externally (i.e., using an external policy server)
- Whether or not internal media policing is turned on or off

About Exact Bandwidth Definition

You can set the amount of bandwidth the Net-Net SBC polices internally and externally in the media profile configuration.

- The Police rate parameter allows you to control the amount of bandwidth the Net-Net SBC polices internally. This value sets the amount of bandwidth to which the Net-Net SBC refers when it allocates resources for a session.
- The Maximum bandwidth allowed per flow (bytes/sec) parameter allows you to control the amount of bandwidth the Net-Net SBC communicates to external policy servers for those server to police or apply allocation constraints.

However, if you configure your Net-Net SBC to police media and leave the Police rate set to zero (its default), the Net-Net SBC refers to the Maximum bandwidth allowed per flow (bytes/sec) value. If it finds a non-zero value set for the Maximum bandwidth allowed per flow (bytes/sec), Net-Net SBC applies this setting for internal policing.

Configuring your Net-Net SBC to police media and leaving the Police rate and the Maximum bandwidth allowed per flow (bytes/sec) set to zero (their defaults) is the same as turning media policing off. Despite the fact that you enabled media policing, the zero values are invalid value the system cannot police.

Internal Media Policing On-Off Switch

To separate internal and external media policing, you can now turn off internal media policing when needed. In previous releases, correctly configuring external bandwidth policing causes your Net-Net SBC to perform both internal and external bandwidth policing.

The Media control tab now contains the Internal media policing EMS label, which acts as a switch to turn media policing on and off. By default, internal media policing is turned on (i.e., this parameter is enabled).

Setting Exact Bandwidth and Disabling Media Policing

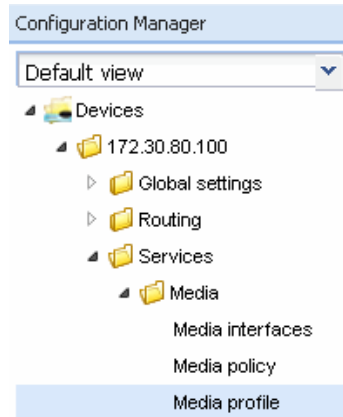
This section shows you how to set exact bandwidth definition and how to turn off internal media policing

Configuring Exact Bandwidth Mapping

You configure exact bandwidth mapping, both internal and external, in the Media profile tab.

To configure exact bandwidth mapping:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click the media profile name in the table and click **Edit**.
4. Scroll to Policing.
5. **Police rate (bps)**—Enter the maximum allowed bandwidth in bytes per second for internal media policing. The minimum and default value for this parameter is 0, and the maximum is 4294967295.

Police rate (bps)

If you configure your Net-Net SBC to police media and this parameter is set to zero (its default), the Net-Net SBC refers to the **Maximum bandwidth allowed per flow (bytes/sec)** value. If it finds a non-zero value set for the **Maximum bandwidth allowed per flow (bytes/sec)**, Net-Net SBC applies this setting for internal policing. Configuring your Net-Net SBC to police media and leaving the **Police rate** and the **Maximum bandwidth allowed per flow (bytes/sec)** set to zero (their defaults) is the same as turning media policing off. Despite the fact that you enabled media policing, the zero values are invalid value the system cannot police.

6. **Maximum bandwidth allowed per flow (bytes/sec)**—Enter the maximum allowed bandwidth in bytes per second for external media policing. The minimum and default value for this parameter is 0, and the maximum is 4294967295.

Maximum bandwidth allowed per flow
(bytes per second),Media policing

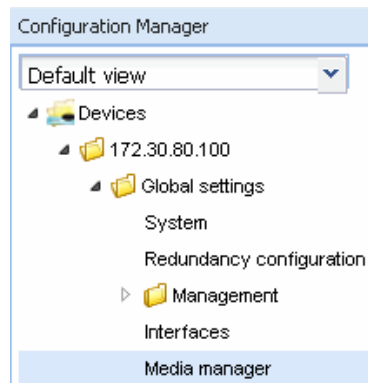
If you configure your Net-Net SBC to police media and the Police rate is set to zero (its default), the Net-Net SBC refers to the **Maximum bandwidth allowed per flow (bytes/sec)** value. If it finds a non-zero value set for the **Maximum bandwidth allowed per flow (bytes/sec)**, Net-Net SBC applies this setting for internal policing.

7. Click **Apply**.

Turning Off Internal Media Policing

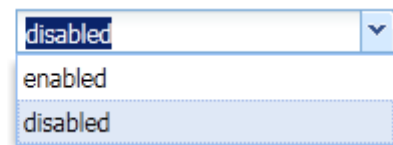
To turn off internal media policing:

1. Expand the Configuration Manager slider > Global Settings.



2. Click Media manager.
3. Click **Add** to add the Media manager configuration in the content area.
4. Click the CPU Policing and DoS tab at the top of the content area.
5. **Internal media policing**—Select enabled from the combination box to uncheck this option, thereby changing this parameter from **enabled** (default) to **disabled**, to turn off internal media policing.

Internal media policing



6. Click **Apply**.

Diameter e2 Realm Identification

This feature applies to Diameter CLF use for e2 interfaces.

For external policy server use with nested realms, the Net-Net SBC allows you to set the realm more precisely when there is a hierarchy of parent-child realm relationships.

In previous releases, when a signaling message arrives on a child realm, the Net-Net SBC uses the same child realm for the Address-Realm in the Globally-Unique-Address AVP in Diameter UDR messages. This behavior is the default. You now have the option of configuring the Net-Net SBC to use the parent realm instead.

How It Works

The Realm location parameter in the External policy server tab specifies whether to use the realm on which a signaling message arrived, or to use that realm's parent.

If you choose to use the parent realm, the Net-Net SBC uses the one associated with the SIP interface on which the REGISTRATION request arrived.

Configuring e2 Realm Identification

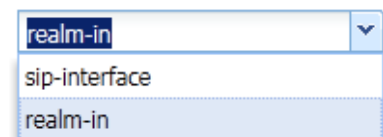
To configure e2 realm identification:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.



2. Click External policy server.
3. Click an external policy server in the table and click **Edit**.
4. **Realm location**—Click realm-in or sip-interface from the combination box to configure the child realm or its parent for the Address-Realm in the Globally-Unique-Address AVL in Diameter UDR messages that the Net-Net SBC sends to the policy server.
 - **realm-in** (default)—The Net-Net SBC uses the same realm upon which the REGISTRATION request arrived.
 - **sip-interface**—The Net-Net SBC uses the realm associated with the SIP interface upon which the REGISTRATION request arrived.

Realm location



5. Click **Apply**.

Diameter Permit on Reject

All Diameter-based interfaces support the ability to allow sessions to proceed through the Net-Net SBC when an external policy server has explicitly rejected the session. This feature provides network administrators with a troubleshooting tool that lets the Net-Net SBC send requests and allows the policy server to respond without call progress being impacted, avoiding a service outage.

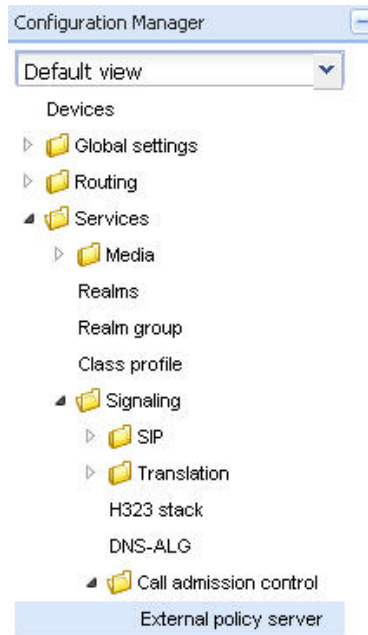
Conditions for Reject

When the Net-Net SBC receives an AAA from the external policy server, it checks for both the Result-Code AVP (268) and Experimental-Result AVP (297) to see if the request has been denied or rejected. If the response indicates success, the call proceeds. If the response indicates that the requests has been denied or rejected and the Permit on Reject feature is enabled, the call still proceeds. Note of allowing a failed call is made in the log file.

Configuring the Net-Net SBC to Permit Rejected Calls

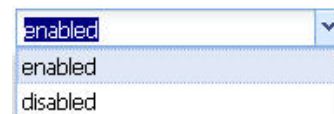
To configure the Net-Net SBC to permit calls rejected by an external policy server:

1. Expand the Configuration Manager slider-> Services-> Signaling-> Call admission control.
2. Click External policy server.



3. Click the external policy server you want to configure in the table and click **Edit**.
4. Scroll to Destination realm AVP.
5. **Permit on reject**—Click **enabled** in the drop down list to allow rejected calls to proceed as if they were allowed.

Permit on reject



6. Click **Apply**.

TACACS+

TACACS+ (Terminal Access Controller Access Control System Plus) is a protocol originally developed by Cisco Systems, and made available to the user community by a draft RFC, *TACACS+ Protocol, Version 1.78 (draft-grant-tacacs-02.txt)*. TACACS+ provides AAA (Authentication, Authorization, and Accounting) services over a secure TCP connection using Port 49.

Introduction

Like DIAMETER and RADIUS, TACACS+ uses a client/server model in which a Network Access Server (NAS) acts in the client role and a TACACS+ equipped device (a *daemon* in TACACS+ nomenclature) assumes the server role. For purposes of the current implementation, the Acme Packet Net-Net SBC functions as the TACACS+ client. Unlike RADIUS, which combines authentication and authorization, TACACS+ provides three distinct functions to provide finer grade access control.

Authentication is the function that confirms a user's purported identity. Authentication is most often based on a simple username/password association, but other, and more secure methods, are becoming more common. The current TACACS+ implementation supports the following authentication methods: simple password, PAP (Protocol Authentication Protocol), and CHAP (Challenge Handshake Authentication Protocol).

Authorization is the function that confirms user privileges. TACACS+ can provide extremely precise control over access to system resources. In the current implementation, TACACS+ controls access to system administrative functions.

Accounting is the function that records what a user does. Depending upon the environment, TACACS+ accounting can serve two purposes. It may be used to account for service access in a billing environment, or it may be used as an auditing tool for security services, as in the current implementation.

TACACS+ provides secure communication between the client and daemon by encrypting all packets. Encryption is based on a shared-secret, a string value known only to the client and daemon. Packets are encrypted in their entirety, save for a common TACACS+ header.

The cleartext header contains, among other fields, a version number, a sequence number, and a session ID. Using a methodology described in Section 5 of the TACACS+ draft RFC, the sender encrypts outbound cleartext messages by repetitively running the MD5 hash algorithm over the concatenation of the session ID, shared-secret, version number, and sequence number values, eventually deriving a virtual one-time-pad of the same length as the message body. The sender encrypts the cleartext message with an XOR (Exclusive OR) operation, using the cleartext message and virtual one-time-pad as inputs.

The message recipient, who possesses the shared-secret, can readily obtain the version number, sequence number, session ID, and message length from the cleartext header. Consequently, the recipient employs the same methodology to derive a virtual one-time-pad identical to that derived by the sender. The recipient decrypts the encrypted message with an XOR operation, using the encrypted message and virtual one-time-pad as inputs.

The TACACS+ implementation is based upon the following internet draft:

draft-grant-tacacs-02.txt, *The TACACS+ Protocol Version 1.78*

Other relevant documents include

RFC 1321, *The MD-5 Message Digest Algorithm*

RFC 1334, *PPP Authentication Protocols*.

RFC 1994, *PPP Challenge Handshake Authentication Protocol (CHAP)*

Authentication

The Net-NET SBC uses TACACS+ authentication services primarily for the authentication of user accounts. Administrative accounts are usually authenticated locally by the SBC, but can be authenticated by TACACS+ services.

The current TACACS+ implementation supports three types of authentication: simple password (referred to as *ascii* by TACACS+), PAP, and CHAP.

ascii Login

ascii login is analogous to logging into a standard personal computing device. The initiating peer is prompted for a username, and, after responding, is then prompted for a password.

PAP Login

PAP is defined in RFC 1334, *PPP Authentication Protocols*. This protocol offers minimal security in that passwords are transmitted as unprotected cleartext. PAP login differs from ascii login in that the username and password are transmitted to the authenticating peer in a single authentication packet, as opposed to the two-step prompting process used in ascii login.

CHAP Login

CHAP is defined in RFC 1994, *PPP Challenge Handshake Authentication Protocol*. CHAP is a more secure than PAP in that it is based on a shared-secret (known only to the communicating peers), and therefore avoids the transmission of cleartext authentication credentials. CHAP operations can be summarized as follows.

After a login attempt, the initiator is tested by the authenticator who responds with a packet containing a *challenge value* — an octet stream with a recommended length of 16 octets or more. Receiving the challenge, the initiator concatenates an 8-bit identifier (carried within the challenge packet header), the shared-secret, and the challenge value, and uses the shared-secret to compute an MD-5 hash over the concatenated string. The initiator returns the hash value to the authenticator, who performs the same hash calculation, and compares results. If the hash values match, authentication succeeds; if hash values differ, authentication fails.

Authentication Message Exchange

All TACACS+ authentication packets consist of a common header and a message body. Authentication packets are of three types: START, CONTINUE, and REPLY.

START and CONTINUE packets are always sent by the Net-Net SBC, the TACACS+ client. START packets initiate an authentication session, while CONTINUE packets provide authentication data requested by the TACACS+ daemon. In response to every client-originated START or CONTINUE, the daemon must respond with a REPLY packet. The REPLY packet contains either a decision (pass or fail), which terminates the authentication session, or a request for additional information needed by the authenticator.

Authorization

The Net-Net SBC uses TACACS+ services to provide administrative authorization. With TACACS+ authorization enabled, each individual ACLI command issued by an *admin* user is authorized by the TACACS+ authorization service. The SBC replicates each ACLI command in its entirety, sends the command string to the authorization service, and suspends command execution until it receives an authorization response. If TACACS+ grants authorization, the pending command is executed; if authorization is not granted, the SBC does not execute the ACLI command, and displays an appropriate error message.

The daemon's authorization decisions are based on a database lookup. Data base records use regular expressions to associate specific command string with specific users. The construction of such records is beyond the scope of this document.

Authorization Message Exchange

All TACACS+ authorization packets consist of a common header and a message body. Authorization packets are of two types: REQUEST and RESPONSE.

The REQUEST packet, which initiates an authorization session, is always sent by the Net-Net SBC. Upon receipt of every REQUEST, the daemon must answer with a RESPONSE packet. In the current TACACS+ implementation, the RESPONSE packet must contain an authorization decision (pass or fail). The exchange of a single REQUEST and the corresponding RESPONSE completes the authorization session.

Accounting

The Net-Net SBC uses TACACS+ accounting to log administrative actions. With accounting enabled, each individual ACLI command executed by an *admin* user is logged by the accounting service.

Accounting Message Exchange

All TACACS+ accounting packets consist of a common header and a message body. Accounting packets are of two types: REQUEST and REPLY.

The REQUEST packet has three variant forms. The START variant initiates an accounting session; the STOP variant terminates an accounting session; the WATCHDOG variant updates the current accounting session. REQUEST packets are always sent by the Net-Net SBC. Upon receipt of every REQUEST, the daemon must answer with a REPLY packet.

A TACACS+ accounting session proceeds as follows:

1. Immediately following successful authorization of an *admin* user, the Net-Net SBC sends an accounting REQUEST START packet.
2. The daemon responds with an accounting REPLY packet, indicating that accounting has started.
3. For each ACLI command executed by an *admin* user, the Net-Net SBC sends an accounting REQUEST WATCHDOG packet requesting accounting of the ACLI command. As the SBC sends the WATCHDOG only after an *admin* user's access to the ACLI command is authorized, the accounting function records only those commands executed by the user, not those commands for which authorization was not granted.
4. The daemon responds with an accounting REPLY packet, indicating that the ACLI operation has been recorded by the accounting function.
5. Steps 3 and 4 are repeated for each authorized ACLI operation.
6. Immediately following logout (or timeout) of an *admin* user, the Net-Net SBC sends an accounting REQUEST STOP packet.

7. The daemon responds with an accounting REPLY packet, indicating that accounting has stopped.

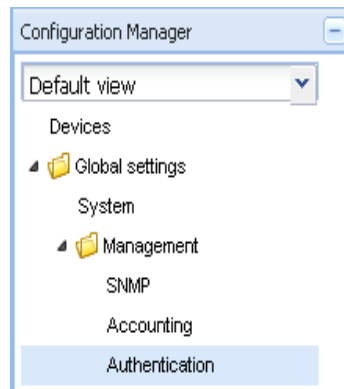
TACACS+ Configuration

Configuration of TACACS+ consists of the following steps:

1. Enable TACACS+ client services
2. Specify one or more TACACS+ servers (daemons)

Enabling TACAS+ Client Services

1. Expand the Configuration Manager slider -> Global settings -> Management.
2. Click Authentication.



3. Scroll to Local RADIUS parameters.
4. **Authentication model**—Select **tacacs** from the drop-down list to enable the TACACS+ AAA protocol.
5. **Authentication protocol**—Select the method used to authenticate Net-Net SBC user accounts from the drop-down list. Supported methods are:
 - ascii* (the default) simple login, Net-Net SBC prompts user for account name and password
 - pap* similar to ascii method, but username and password are encapsulated in a PAP header
 - chap* authentication based on a shared-secret, which is not passed during the authentication process
6. Scroll to TACAS.
7. **TACACS plus authorization**—Leave this attribute enabled for command-based authorization of *admin* users. By default, authorization is enabled.
8. **TACACS+ accounting**—Select enabled from the drop-down list to enable accounting *admin* operations. By default, accounting is enabled.
9. **Server assigned privilege**—Select enabled from the drop-down list to use a proprietary protocol, which must be supported by both the TACACS+ client and daemon, that provides the Net-Net SBC with explicit feedback identifying the

privilege level (*user* or *admin*) assigned by the TACACS+ daemon to an authenticated account.

The default behavior, that provides the Net-Net SBC with a binary (passed authentication/failed authentication) result, is described in the [Authentication](#) section.

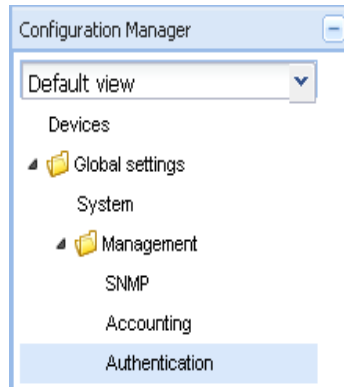
With **Server assigned privilege** set to **enabled**, a successful authentication is followed by an additional proprietary message exchange that informs the Net-Net SBC of the privilege level assigned to the newly authenticated account. If the account is assigned *user* privileges, the Net-Net SBC denies access to the ACLI **enable** command; if the account is assigned *admin* privileges, access to the ACLI enable command is granted.

10. Click **Apply**.

Specify TACACS+ Servers

Use the following procedure to specify one or more TACACS+ servers (daemons).

1. Expand the Configuration Manager slider -> Global settings -> Management.
2. Click Authentication.



3. Scroll to the TACACS plus server table and click **Add**.
4. **IP Address**—Enter the the IP address of the TACACS+ daemon.
5. Click **Apply**. A Success dialog box appears.
6. Click **OK**.
7. Click the IP address of the TACACS+ server you want to configure and click **Edit**.
8. **Realm name**—Select the realm from the drop-down list.
9. **Out of service period(sec)**—Specify, in seconds, the length of the quarantine period imposed upon TACACS+ daemons that become unreachable. Quarantined servers are not eligible to participate in the server-selection algorithm
Supported values are within the range 10 through 10000 seconds, with a default value of 10.
10. Click **OK**.
11. Select the Tacacs server configuration from the table and click **Edit**.
12. **Administrative state**—Click the checkbox to enable the availability of this TACACS+ daemon.

Only TACACS+ daemon that are in the enabled state are considered when running the server-selection algorithm.

13. **Port**—Enter the daemon port that receives TACACS+ client requests.

Provide a port number within the range 1025 through 65535, or retain the default value, 49, the *well-known* TACACS+ port.

14. **Secret**—Provide the shared-secret used by the TACACS+ client and the daemon to encrypt and decrypt TACACS+ messages. The identical shared-secret must be configured on associated TACACS+ clients and daemons.

Enter a 16-digit string, and ensure that the identical value is configured on the TACACS+ daemon.

15. **Supported authentication method**—Select the method used to authenticate Net-Net SBC user accounts from the drop-down list. Supported methods are:

ascii (the default) simple login, Net-Net SBC prompts user for account name and password

pap similar to *ascii* method, but username and password are encapsulated in a PAP header

chap authentication based on a shared-secret, which is not passed during the authentication process

16. Click **Apply**.

Managing TACAS+ Operations

TACACS+ management is supported by the following utilities.

TACACS+ MIB

An Acme Packet proprietary MIB provides SNMP-based access to TACACS+ statistics.

MIB counters are contained in the apSecurityTacacsPlusStatsTable that is defined as follows.

```
SEQUENCE {
    apSecurityTacacsPlusCliCommands          Counter32
    apSecurityTacacsPlusSuccess Authentications Counter32
    apSecurityTacacsPlusFailureAuthentications Counter32
    apSecurityTacacsPlusSuccess Authorizations Counter32
    apSecurityTacacsPlusFailureAuthorizations Counter32
}
```

Object Identifier Name: apSecurityTacacs Table (1.3.6.1.4.1.9148.3.9.1.4)		
Object Identifier Name: apSecurityTacacsEntry (1.3.6.1.4.1.9148.3.9.1.4.1)		
apSecurityTacacsIndex	apSecurityTacacsEntry: 1.3.6.1.4.1.9148.3.9.1.4.1.1	Unique index for the tacacs table
apSecurityTacacsServer	apSecurityTacacsEntry: 1.3.6.1.4.1.9148.3.9.1.4.1.2	The server for the last transaction
apSecurityTacacsCliCommands	apSecurityTacacsEntry: 1.3.6.1.4.1.9148.3.9.1.4.1.3	Number of CLI commands sent for TACACS+ accounting
apSecurityTacacsSuccessAuthentication	apSecurityTacacsEntry: 1.3.6.1.4.1.9148.3.9.1.4.1.4	Number of successful TACACS+ authentication requests
apSecurityTacacsFailureAuthentication	apSecurityTacacsEntry: 1.3.6.1.4.1.9148.3.9.1.4.1.5	Number of failed TACACS+ authentication requests
apSecurityTacacsSuccessAuthorization	apSecurityTacacsEntry: 1.3.6.1.4.1.9148.3.9.1.4.1.6	Number of successful TACACS+ authorization requests
apSecurityTacacsFailureAuthorization	apSecurityTacacsEntry: 1.3.6.1.4.1.9148.3.9.1.4.1.7	Number of failed TACACS+ authorization requests

SNMP Trap

SNMP traps are issued when

- a TACACS+ daemon becomes unreachable
- an unreachable TACACS+ daemon becomes reachable
- an authentication error occurs
- an authorization error occurs
- an accounting error occurs

TACACS+ Logging

All messages between the Net-Net SBC and the TACACS+ daemon are logged in a cleartext format, allowing an *admin* user to view all data exchange, except for password information.

Diameter: Subscriber Information Enhancement

1775

You can configure the Net-Net SBC to send the Subscription-Id AVP in the AAR message destined for policy servers. The Subscription-Id AVP contains the user's URI information, and is used in 3GPP Rx reference points. You might need to configure this functionality if your network's policy servers rely on having this information to identify endpoints/subscribers. Without this function enabled, the Net-Net SBC provides the L3 IP address, which is not enough information for some policy servers.

About the Subscription-Id AVP

The Subscription-Id AVP (code 443; type is Grouped) identifies the end-user's subscription and contains the following:

- Subscription-Id-Type AVP—(code 450; type is enumerated) Defines the subscription type as assigned by IANA; the server must implement all types required to perform credit authorization for the services it offers, including possible future values. Subscription-Id-Types must be created according to the "M" flag rule.
 - END_USER_E164 (0)—International E.164 format
 - END_USER_IMSI (1)—International IMSI format
 - END_USER_SIP_URI (2)—SIP URI format
 - END_USER_NAI (3)—Network Access Identifier format
 - END_USER_PRIVATE (4)—Credit-control server private identifier
- Subscription-Id-Data AVP—(code 444; type is UTF8String) Identifies the end user. The Net-Net SBC populates this AVP with the subscriber's Request-URI corresponding to the AAR message. In other words, the Subscriber_id URI corresponds to the currently supported IP address of the subscriber included in the Frame-IP-address AVP.

If the Request-URI is of the type:

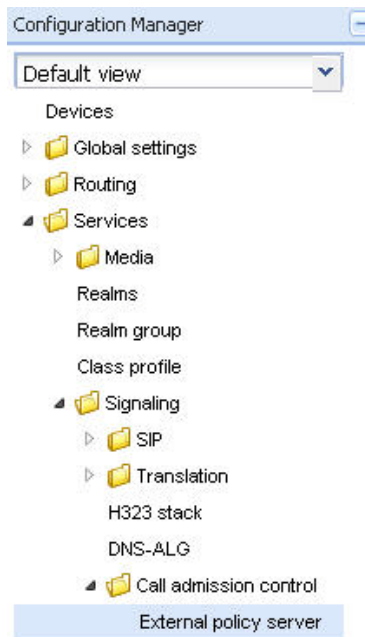
- Tel URI, the Subscription-Id-Type AVP is END_USER_E164
- SIP or SIPS URI, the Subscription-Id-Type AVP is END_USER_SIP_URI

Currently, the Net-Net SBC supports SIP, SIPS, and Tel URIs. So the Net-Net SBC sends out the Subscription-Id AVP only if the Request-URI received is one of these three types.

Configuring Subscription-Id AVP Support

To add Subscription-Id AVP support to your external policy server configuration:

1. Expand the Configuration Manager slider-> Services -> Signaling -> Call admission control.
2. Click External policy server.

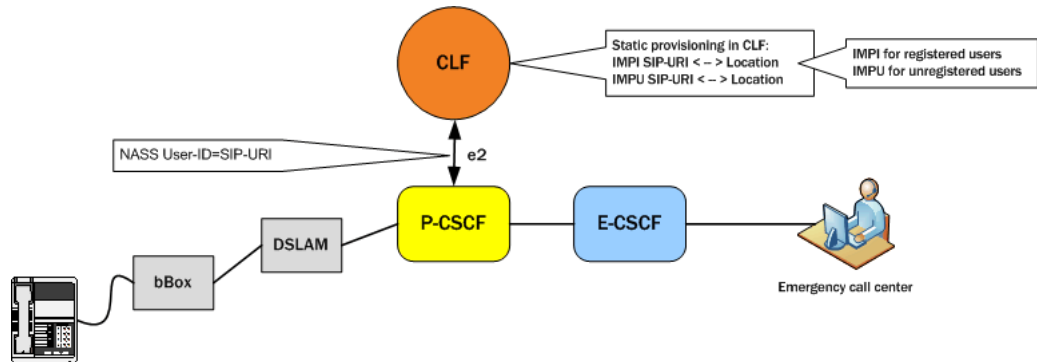


3. Click the External policy server you want to configure in the table and click **Edit**.
4. Click **Options** next to Optional AVPs. The Options window appears.
5. Click **include-sub-info** in the Available options list to select it.
6. Click  to move the option to the Selected options list. The option name appears in the Selected options list along with a description in the lower part of the window.
7. Click **OK**.
8. Click **Apply**.

DIAMETER: CLF e2 Interface User-Name AVP Support

In compliance with ETSI ES 283 -35 V1.2.1, the Net-Net SBC's e2 interface can query the CLF using the SIP endpoint's IP address or its NASS User-ID. The system's e2 interface uses the SIP-URI to query the CLF by including the User-Name AVP in the User Data Request (UDR). The CLF can then furnish the P-CSCF (i.e., the Net-Net SBC) with the INSEE code in the Location-Information AVP in its User Data Answer (UDA).

This diagram shows how this capability works. The Net-Net SBC acts as the P-CSCF.



How It Works

This section describes how the Net-Net SBC handles CLF e2 interface User-Name AVP support for registering and unregistered users.

- **Registering users**—When it receives a registration request, the Net-Net SBC checks the incoming SIP interface to determine CLF use. If CLF use is unnecessary, the Net-Net SBC forwards the registration message to its destination.

When CLF is required, the Net-Net SBC selects the AVPs to send the CLF, including the User-Name AVP before sending it a UDR. A **none** setting for this parameter means the Net-Net SBC does not include the User-Name AVP in any UDRs. The Net-Net SBC adds the User-Name AVP to the UDR if the **user-name-mode** parameter in the external policy server configuration is set to:

- **endpoint-id**—IP address of the registering endpoint is sent as the payload for the User-Name AVP
 - **public-id**—SIP-URI portion of the TO header from the register message is sent as the payload for the User-Name AVP
 - **auth-user**—Username attribute of the Authorization header from the register is sent as the payload for the User-Name AVP; if there is no authorization header, the Net-Net SBC will not consult the CLF and will forward the registration message
- **Unregistered users**—When it receives an INVITE request, the Net-Net SBC checks the incoming SIP interface to determine if it should use an external policy server. If it does not need to use an external policy server, the Net-Net SBC forwards the INVITE message to its destination.

When the Net-Net SBC does need to use an external policy server, it also checks to determine if the INVITE is in a registration and if location data in the registration cache is available for that endpoint. These requirements being met, the Net-Net SBC inserts the P-Access-Network-Info header with the location

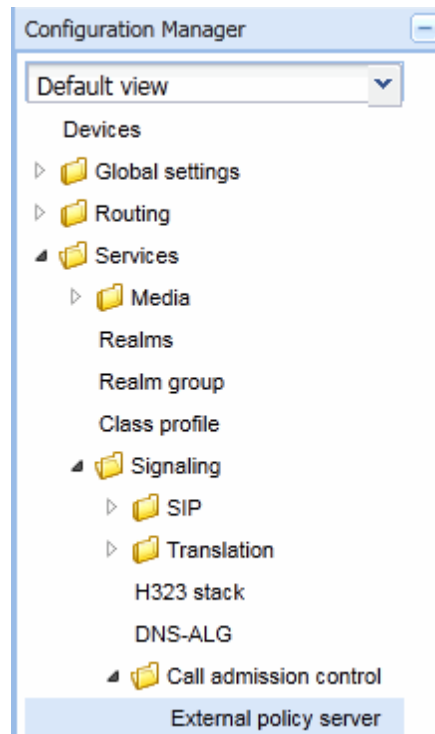
string into the INVITE it forwards to the destination. If these requirements are not met, the Net-Net SBC consults the CLF before forwarding the INVITE. The following describe the impact of the user-name-mode setting in such instance:

- **endpoint-id**—IP address of the endpoint that issued the INVITE is sent as the payload for the User-Name AVP
- **public-id** and **auth-user**—SIP-URI portion of the INVITE is sent as the payload for the User-Name AVP:
 - a) With a P-Asserted-Id header present in the INVITE, the Net-Net SBC uses the first PAID header (if there are multiple PAID headers)
 - b) Without a P-Asserted-Id header present in the INVITE's P-Preferred-Identity header, the Net-Net SBC uses the first PPI (if there are multiple PPI headers)
 - c) With neither P-Asserted-Id nor P-Preferred-Identity header present, the Net-Net SBC uses the From header

Configure DIAMETER Permit on Reject

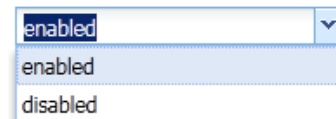
To enable **DIAMETER permit on reject**:

1. Expand the Configuration Manager slider -> Services -> Signaling -> Call admission control.



2. Click External policy server.
3. Select the external policy server you want to use and click **Edit**.
4. **Permit on Reject**— Select Enabled from the combination box to allow

Permit on reject



5. Click **Apply**.

Diameter based RACF & CLF IPv6 Support

1853

The Net-Net SBC supports Diameter-based CLF and RACF external policy servers in both IPv4 and IPv6 networks. There are three areas of enhanced functionality where the Net-Net SBC's Diameter external policy server offerings have changed.

1. AVP support of IPv6 addresses encoded in UTF-8 format
2. Extra bandwidth allocation in policed flows to compensate for longer addresses
3. Framed-IPv6-Prefix AVP (97) support

IPv6 Addresses in UTF-8 Format

When necessary, the Net-Net SBC checks that IPv6 addresses are formatted correctly in UTF-8 when they are inserted into relevant AVPs. The applicable AVPs are

- Flow-Description AVP (507)
- Subscription-Id-Data AVP (444)
- Destination-Realm AVP (283) only if applicable

Framed-IPv6-Prefix AVP

The Net-Net SBC supports the Framed-IPv6-Prefix AVP (97) as defined in RFC4005.

The Diameter interface on the Net-Net SBC substitutes this AVP for the Framed-IP-Address AVP (8) in a Diameter message when carrying IPv6 addresses in the AVP.

The IP Address is encoded as Octet-String. Although the IPv6 address is a 128 bit number, it will not fall on a 4 byte boundary due to the formatting of this particular AVP. Additional whitespace will be added per RFC 3588 to the end of the octet-string to pad the ending of the AVP.

Bandwidth Allocation Compensation for IPv6

Transporting IPv6 packets requires extra bandwidth because of their larger packet size. This needs to be taken into account when allocating bandwidth and policing media. In order to reserve the necessary bandwidth for signaling messages, the **standard packet rate** parameter has been added to the *media profile* configuration element.

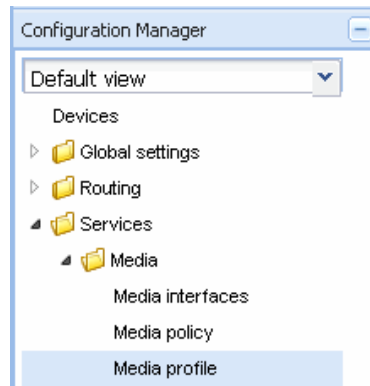
The Net-Net SBC needs a baseline media packet size for bandwidth requests. It first checks if the SDP includes a *ptime* value. That value will be used as the baseline if present in the request. If this value is not included in the SDP, the **standard packet rate** parameter is used as the baseline. The chosen value is then multiplied times 20 (the difference between an IPv4 and IPv6 packet).

This value is sent to the external policy server in the bandwidth request when sending media into an IPv6 realm. From that point it is used when allocating bandwidth or media policing.

Adding a Standard Packet Rate

To add a standard packet rate for a media profile:

1. Expand the Configuration Manager slider -> Services -> Media.
2. Click Media profile.



3. Click the media profile you want to configure in the table and click **Edit**.
4. **Standard packet rate**—Enter the value to use for the standard packet rate for this codec when sending a request to the RACF for a bandwidth request.
 Standard packet rate
5. Click **Apply**.

Signaling Only P-CSCF

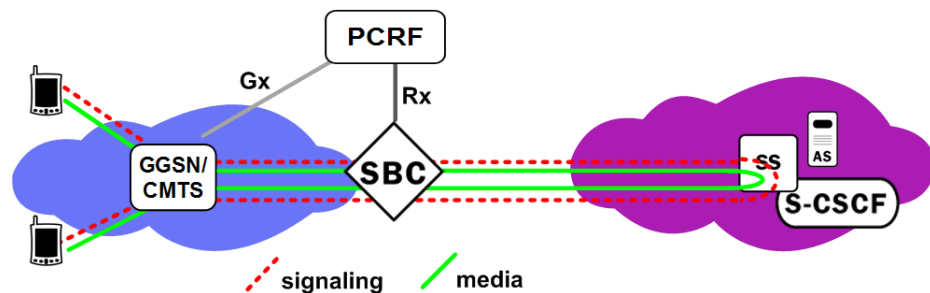
When configured as a P-CSCF, the Net-Net SBC can release media between two endpoints if the network topology allows it. Please refer to the following sections in the Realms and Nested Realms chapter to review the conditions and configurations required for media release:

- Media Release Across SIP Network Interfaces
- Media Release Behind the Same IP Address
- Media Release between Endpoints with the Same IP Address
- Media Release Behind the Same NAT IP Address

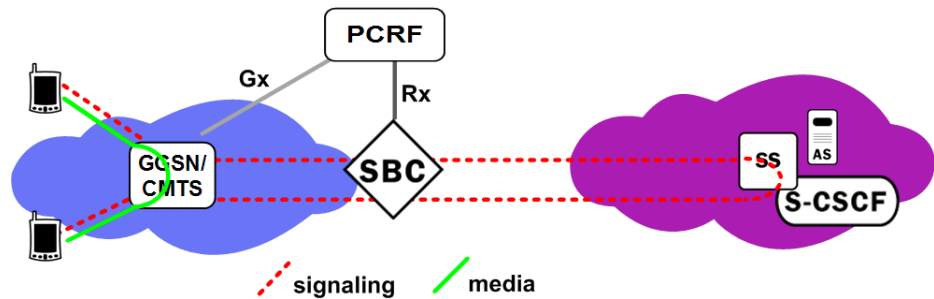
Media release with external policy server control is available when the Net-Net SBC, acting as a P-CSCF, connects via the Rx interface to a PCRF.

Media Path Deployment Models

After two UAs' session has been set up, the media follows the same path as the signaling, through the Net-Net SBC. This is the default managed media behavior.



In the released media scenario, as the call is set up, the Net-Net SBC determines if media release is possible. If it is, media is released between the two endpoints to take the most direct path. The Net-Net SBC manipulates the SDP so media is exchanged directly between the endpoints. While the signaling flows traverse the Net-Net SBC and other network elements, the media flows are routed or forwarded between the two endpoints.



Managed Media Advantages

The following applications are possible when the Net-Net SBC manages media:

- The Net-Net SBC can monitor media by maintaining flow guard timers. It also has tools to enforce bandwidth constraints on media streams.
- The Net-Net SBC can perform topology hiding by obfuscating core IP addresses between UAs.
- The Net-Net SBC can protect against signaling and bandwidth-based attacks.
- The Net-Net SBC can perform lawful interception on both signaling and media.

Released Media Characteristics

The following applications are possible when the Net-Net SBC releases media between UAs:

- When deployments include a MRFP and/or MGW, they need to monitor media flows and have access to endpoints' IP addresses; this is only possible with released media.
- The Net-Net SBC can protect against signaling-based DoS attacks.
- The Net-Net SBC can perform lawful interception only on signaling flows.

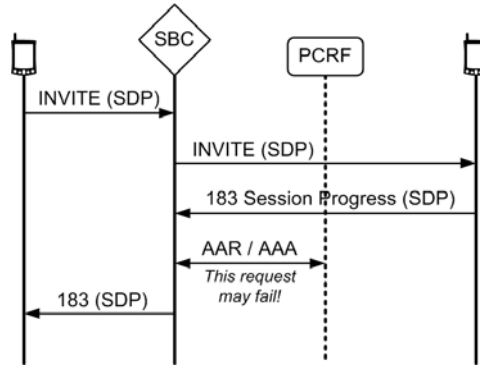
Bandwidth Request Models

After the Net-Net SBC identifies that a call can have its media released between endpoints, it refers to configuration for how to request bandwidth from the external policy server when setting up the call. The Net-Net SBC can use one of three models for requesting bandwidth and authorization from the external policy server.

Bandwidth Request Upon SDP Answer

In this model, fully specified policy server requests occur: the AAR message contains all final media and signaling information about the call, as gleaned from the SDP answer. Resources are requested and confirmed by the policy server after the SDP

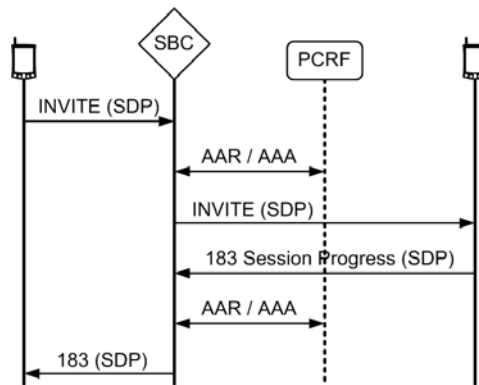
negotiation succeeds. In some cases, the callee UA may ring and subsequently be disconnected. The following is a simplified call flow.



This is the default behavior of the Net-Net SBC and occurs when the **reserve incomplete** parameter is left to its default, **disabled**.

Bandwidth request Upon SDP Offer and Answer

In this model, partially specified policy server requests occur as the call is set up. When preliminary information is sent to the policy server for authorization, the AAR includes the Service-Info-Status AVP (code 527) with a value of 1 that indicates further negotiation will occur. Thus, resources are requested and confirmed by the policy server before ringing the callee, then once again after the SDP negotiation succeeds. Reserved resources may be modified based on SDP answer depending on selected codec. The following is a simplified call flow.



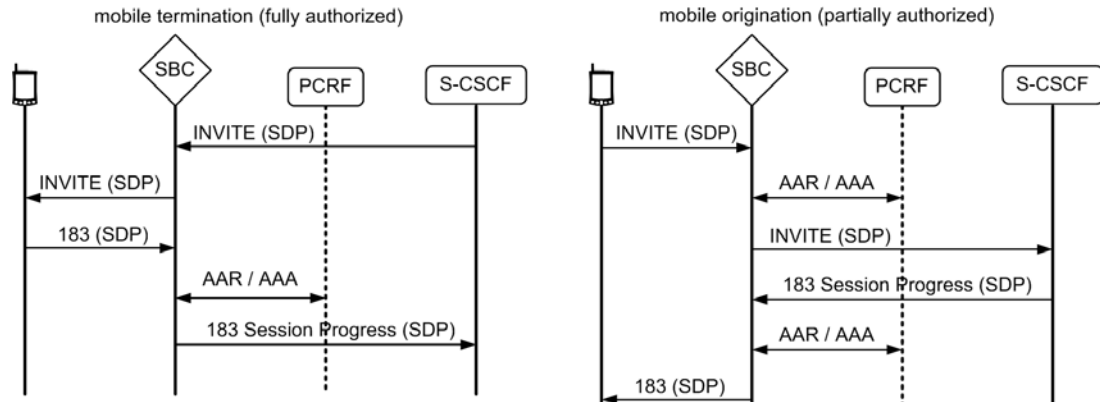
This behavior occurs when the **reserve incomplete** parameter is set to **enabled**.

Hybrid Bandwidth Request Model

This model, a hybrid of the first two, is useful in mobile scenarios. Some PCRF/PDF applications require the UA's location in order to successfully respond to an AAR query. When a UA has been idle for an extended period of time, its location needs to be refreshed by a page to determine the servicing ASN. This page typically takes the form of an L3 packet. As the phone's location has been refreshed, the Net-Net SBC can forward the INVITE and make a fully qualified AAR request that includes the location; none of the information in the AAR is provisional.

At the same time, when the call originates in the access network, the endpoint's location is known to the network upon receipt of the INVITE - the endpoint left its idle state. Since all expected information is known, the Net-Net SBC can make a

provisional AAR query to the external policy server like in the Bandwidth request Upon SDP Offer and Answer.



This behavior occurs when the **reserve incomplete** parameter is set to **orig realm only**.

Configuring External Policy Server's Media Release Behavior

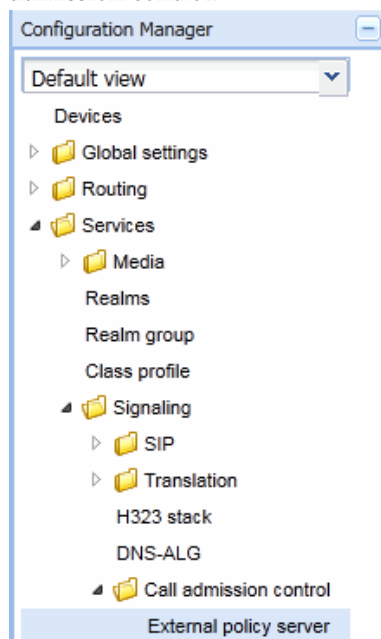
The Net-Net SBC must first allow media release, which is configured by realm. Ensure that you have at least one of the following realm parameters configured to enabled:

- mm-in-realm
- mm-in-network
- mm-in-system
- mm-same-ip

There is no external policy server configuration associated with this feature.

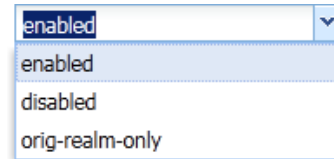
To configure the external policy server's media release behavior:

1. Expand the Configuration Manager slider -> Services -> Signaling -> Call admission control.



2. Click External policy server.
3. **Early resource reservation**—Set this parameter to one of the following values for the stated use:
 - **disabled**—bandwidth request upon SDP answer
 - **enabled**—bandwidth request upon SDP offer and answer
 - **orig-realm-only**—hybrid model of using bandwidth requests on SDP answer and SDP offer and answer depending on the call's origination from core or access realm.

Early resource reservation



4. Click Apply.

Asynchronous SIP-Diameter Communication

The Net-Net SBC's Diameter-based external policy server support offers an asynchronous mode which can address the following two situations:

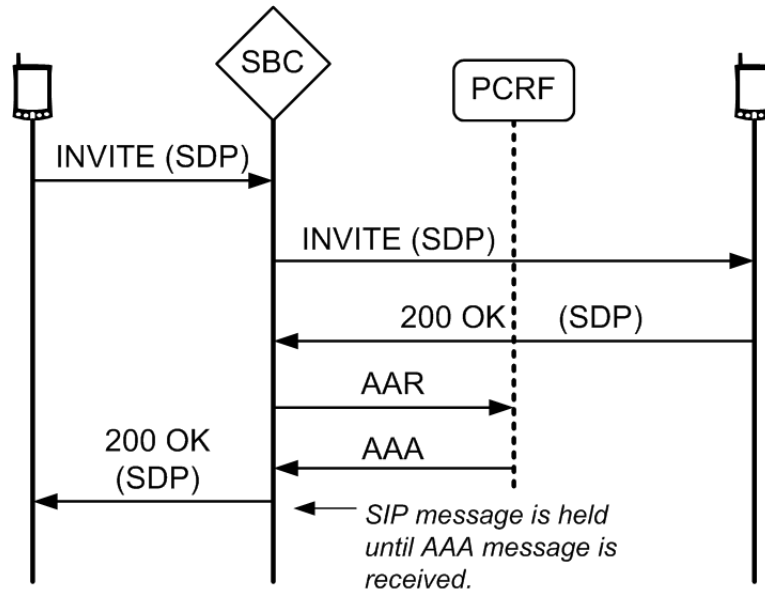
One of the fundamental behaviors in the Net-Net SBC's Diameter model relies on the external policy server making an authorization decision which is then communicated to the Net-Net SBC. Part of the call authorization sequence of events involves the Net-Net SBC waiting for an external policy server's response before the Net-Net SBC can completely set up, modify, or update the call. The long pause that the endpoints experience, while the Net-Net SBC holds up SIP flows waiting for the external policy server's response, may lead to unnecessary call failure situations.

In some Diameter-based external policy server deployments, the media traverses a CMTS at the edge of the network; the CMTS gates may be established by a policy server to dynamically enable QoS to/from a UE toward another UE. If no gate is established then the media should still traverse the CMTS and be admitted to the network with a "best-effort" network path.

Synchronous Model

The Net-Net SBC's default external policy server model may be referred to as the synchronous model; The Net-Net SBC sends a Policy Server request based on a

SIP/SDP trigger point, the SIP signaling is held until a response is returned from the policy sever. For example:



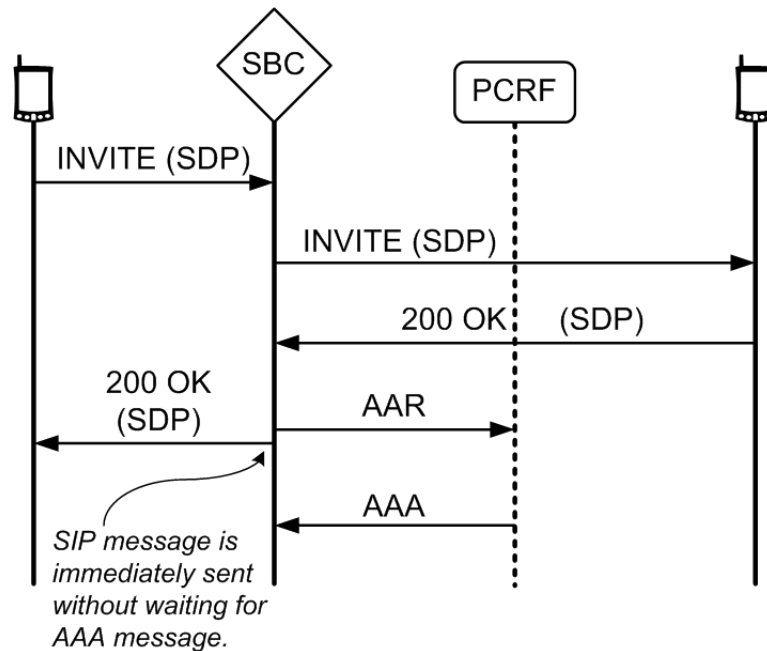
Note: In the above diagram, the **reserve incomplete** parameter has been set to **disabled**.

Asynchronous Model

In the asynchronous model the request that the Net-Net SBC's sends to the Policy Server flows in an asynchronous state with respect to SIP messaging. Thus the Net-Net SBC allows the SIP session to proceed naturally, and does not pause for outstanding Policy Server answers to be received.

It is crucial to understand that the Policy Server's ultimate success or failure answer, or timeout, as related to a SIP session does not impact the establishment of the SIP session itself.

The following diagram represents the asynchronous model where SIP sessions always succeed. SIP signaling is never delayed and as a result, call set-up time is decreased.



Note: In the above diagram, the **Early resource reservation** parameter has been set to **disabled**.

Caution

Acme Packet recommends that for each Net-Net SBC, the same model (Synchronous / Asynchronous) is used for all external policy servers configuration instances. Failure to follow this guideline may result in complex interactions from a timing perspective which may lead to a negative end user experience.

Serialized Diameter Messaging

Access networks with complex policy server structures might introduce the possibility of un-ordering Diameter requests into a given CMTS, even if Diameter message delivery was correctly ordered on the TCP connection between the Net-Net SBC and a lower-tiered external policy server.

The Net-Net SBC can prevent the external policy server from receiving out-of-order messages at the application layer by serializing them. The Net-Net SBC serializes Diameter messages by ensuring a Diameter request for one sessionID is not sent until an answer is received from the previous request for the same sessionID. The Net-Net SBC applies this constraint while waiting for Diameter response or when considering a Diameter request timeout (15 seconds).

Serialized Diameter messaging is always enforced for Diameter-based External Policy Server communication.

High Availability Considerations

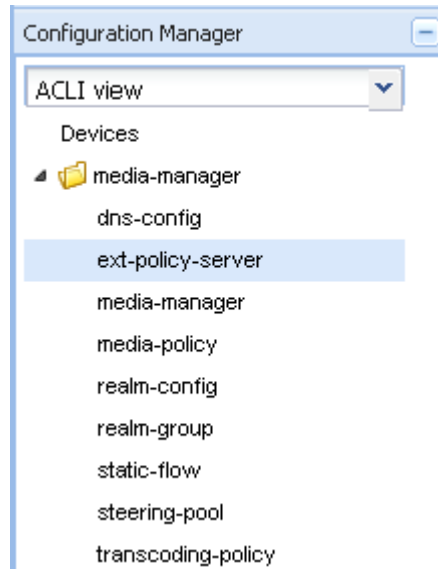
Upon a failover, the Net-Net SBC may not guarantee the synchronicity of Diameter messages for active sessions. For example, if the active Net-Net SBC sent an AAR and an AAA is yet to be received, upon failover the newly active SBC may send a 2nd AAR or an STR without receipt of the initial AAA. This creates an unavoidable race condition. Such rare cases must be handled appropriately within the Quality of Service subsystem (Policy Servers and CMTSs).

Configuring the External Bandwidth Manager

You must configure media profiles to utilize bandwidth policing. This feature can only be enabled in the ACLI view.

To configure the external bandwidth manager to run in asynchronous mode:

1. Expand the Configuration Manager slider.



2. Select ACLI view from the combination box.
3. Expand the media-manager folder and click ext-policy-server.
4. Select the external policy server that you want to use and click **Edit**.
5. **asynchronous-mode**—Select enabled to enable this external policy server to run in asynchronous mode.

asynchronous-mode enabled
6. Click **Apply**.

FQDN as Address for External Policy Server

You can configure an external policy servers' address as an FQDN in addition to an IP address.

FQDN Configuration

FQDNs are configured in the address parameter in the external policy server configuration element. These FQDNs must conform to RFC 1035. If the port parameter in external policy server configuration element is not zero, then it will be used to connect to the group of applicable external policy servers. If the port parameter is set to zero, then the port number returned in SRV RR from the DNS server will be used. When the Net-Net SBC queries a DNS server for an FQDN configured in the external policy server, it always appends the `_diameter._tcp.` to request only TCP based Diameter servers.

Diameter Policy Server High Availability

External Policy Server HA Cluster

The Net-Net SBC can provide external policy server redundancy through a combination of multiple servers being returned in one FQDN query and maintaining state of these servers.

When multiple IP addresses are returned in a response to a DNS query for Diameter-based external policy servers, the Net-Net SBC assembles the IP addresses into an HA cluster which provides redundancy for Diameter-based applications. This feature is enabled by configuring the external policy server's **address** parameter with an FQDN.

The number of servers maintained in the HA cluster is configured with the **max connections** parameter. Thus if the **max connections** parameter is set to 3, the Net-Net SBC maintains 1 active external policy server with 2 back up servers.

Standby Server Prioritization

The Net-Net SBC looks at the priority and weight fields in the DNS response to create the preferred order of the the primary, secondary, tertiary, and quaternary Policy Servers. These 4 addresses are known as the top-level PSs in the cluster.

The Net-Net SBC uses the priority and weights according to RFC 2782. However, weights only apply when multiple servers share the same priority value. If the priorities are different, then the Net-Net SBC will not use weights. If the priorities are the same, then weight of the two (or more) contested servers is used to determine which one to use.

Server States

An HA cluster can contain up to 4 Policy Servers, where the TCP/Diameter connection is established and monitored. Diameter session traffic is only sent to the active policy server in the cluster. The policy servers exist in the following states:

- active-TCP and Diameter connection established. Net-net SBC using this server for policy decisions. The policy server with the highest priority/weight begins in this state.
- standby- TCP and Diameter connection established. Server in standby mode.
- inactive - Diameter connection not successfully established. Net-Net SBC tries to reconnect to inactive servers.

HA Cluster Refresh

The Net-Net SBC sends followup SRV queries to the DNS server to refresh the list of available policy servers in the cluster in the following instances:

- DNS cache expires after the TTL is exceeded
- a new policy server with FQDN for an address is configured, saved, and activated on the Net-Net SBC
- after an SPU switchover, the newly active SPU performs a DNS query

When the Net-Net SBC re-queries the DNS server for Diameter external policy servers, the cluster is refreshed with the new/changed servers. Policy server priority is also reconfigured based on newly returned priorities and weights. Upon a cluster refresh, the Net-Net SBC:

- closes connections with standby policy servers that are no longer in the cluster
- creates connections with policy servers which are new in the set

If the currently active policy server remains a member of the cluster after a refresh, it remains active even if its priority has changed. If the active-before-the-refresh policy server is not a member of the cluster after a DNS refresh, the Net-Net SBC gracefully closes the connection to this server. The Net-Net SBC then installs the highest priority server as the active policy server

DNS Failure

If the Net-Net SBC fails to get a response from the DNS server or does not receive at least one IP address in the SRV RR, it continues to send SRV queries periodically, starting with 5 seconds and doubling the interval for every sequential failure, until it receives a valid response. While waiting for a successful DNS response, the Net-Net SBC uses the existing Diameter servers in the DNS cache.

Policy Server Failover

The active external policy server fails over to the highest priority standby server when:

- the TCP connection is closed due to a RST or FIN
- the Diameter connection (CER/CEA exchange) could not be established
- a configured number of consecutive Diameter message timeouts occur. This number is configured in the **max timeouts** parameter. The max timeouts parameter refers to all Diameter messages except for the following:
- You can configure Diameter keepalive message time-outs separately from all other Diameter messages by setting **watchdog ka timer** in the external policy server configuration element. This will failover the active policy server based on a timeout value for DWR messages.
- You can configure Diameter STR message time-outs separately from all other Diameter messages by setting the **str-retry=<timeout number>** option in the

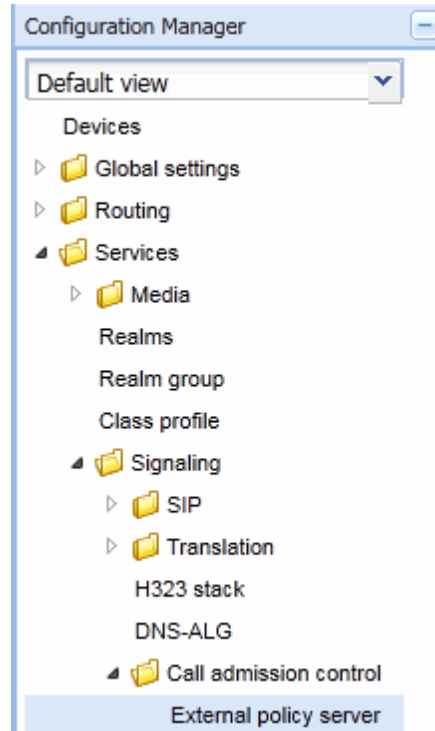
external policy server configuration element. This will failover the active policy server based on a unique timeout value for STR messages.

If the Net-Net SBC sends an AAR/STR message to the active policy server and then switches to a different policy server, any new Diameter messages related to that session are sent to the same policy server as long as it is not inactive. If that server becomes inactive, messages will be sent to the new policy server. The new policy server however will not recognize the sessionID and discard the request.

Configure External Bandwidth Manager

To configure the external bandwidth manager:

1. Expand the Configuration Manager slider -> Services -> Signaling -> Call admission control.



2. Click External policy server.
3. Click the external policy server you would like to use and click **Edit**.
4. **Maximum number of connections**—Set the number of servers to maintain in an external policy server cluster. Valid values range from 1 - 20.

*Maximum number of connections

5. **Maximum message transmission**—Set the number of request timeouts before the Net-Net SBC sets this external policy server to inactive. You can separately set the number of DWR timeouts that trigger a server to be inactive as an option. Valid values range from 0 - 200.

*Maximum message transmission

6. **Selection Strategy**—Leave this parameter at its default, *Failover* setting.

Selection strategy

7. **Address**—Enter the IP address or FQDN of the external policy server. To use external policy server redundancy, you must enter the address as an FQDN.

Address

192.168.1.101

8. **Port number**—If you configure this parameter, it will override the port returned in a DNS reply.

Port number

0

9. Click **Apply**.

Multi-tiered Policy Server Support

When accepting a call in a 2 tiered policy server environment, the Net-Net SBC queries the Tier-1 policy server, running in stateless mode, for authorization. The Tier-1 PS, identifies an appropriate Tier-2 policy server for every incoming DIAMETER session. Once a Tier-2 PS is chosen to control a given session-ID, all corresponding transactions for that session are sent toward the chosen Tier-2 policy server. The mapping of session-ID to Tier-2 PS is performed by either the Tier-1 PS or the P-CSCF. Using a stateless Tier-1 PS in this capacity increases overall performance.

To act in the P-CSCF role with a stateless Tier-1 policy server, the Net-Net SBC recalls the Origin-Host-AVP value returned in the initial AAA response to the Diameter session establishment query. The Origin-Host-AVP value is then inserted into the Destination-Host-AVP in all subsequent Diameter messages (AAR, STR) corresponding to that session-ID.

Messaging Details

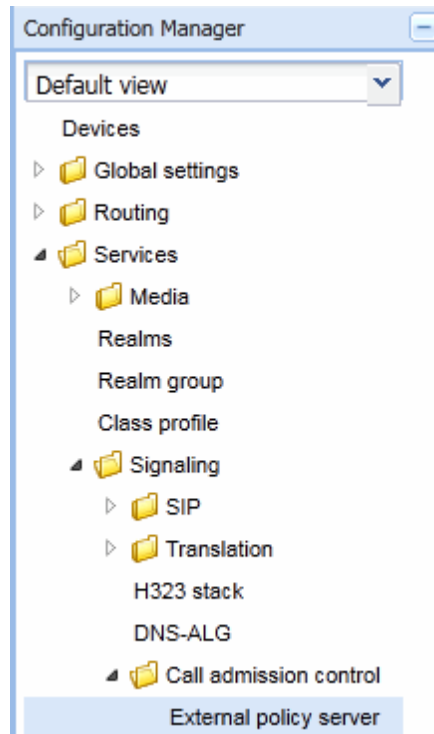
The Net-Net SBC's normal behavior when communicating with a Diameter based policy server is to use the received STR message's Origin-Host AVP, as returned in the CEA, populating an outgoing Destination-Host AVP. Enabling the **cache destination host** parameter inserts the Origin-Host-AVP value from the first AAA message into the Destination-Host-AVP in all subsequent Diameter messages (AAR, STR) corresponding to that session-ID.

If the Origin-Host AVP value, as received by the Net-Net SBC, changes mid session, all subsequent Destination-Host-AVP values are updated with the change.

Multi-Tiered Policy Server Support

To configure Multi-tiered policy server support:

1. Expand the Configuration Manager slider -> Services -> Signaling -> Call admission control.



2. Click External policy server. The External policy server window appears.
3. Select the external policy server you would like to use and click **Edit**.
4. **Cache destination host**—Set this parameter to **enabled** for the Net-Net SBC to participate as a P-CSCF in a multi-tiered policy server environment.

Cache destination host

enabled	▼
enabled	
disabled	

5. Click **Apply**.

FQDN as Address for External Policy Server

You can configure an external policy servers' address as an FQDN in addition to an IP address.

FQDN Configuration

FQDNs are configured in the address parameter in the external policy server configuration element. These FQDNs must conform to RFC 1035. If the port parameter in external policy server configuration element is not zero, then it will be used to connect to the group of applicable external policy servers. If the port parameter is set to zero, then the port number returned in SRV RR from the DNS server will be used. When the Net-Net SBC queries a DNS server for an FQDN configured in the external policy server, it always appends the `_diameter._tcp.` to request only TCP based Diameter servers.

Diameter Policy Server High Availability

External Policy Server HA Cluster

The Net-Net SBC can provide external policy server redundancy through a combination of multiple servers being returned in one FQDN query and maintaining state of these servers.

When multiple IP addresses are returned in a response to a DNS query for Diameter-based external policy servers, the Net-Net SBC assembles the IP addresses into an HA cluster which provides redundancy for Diameter-based applications. This feature is enabled by configuring the external policy server's **address** parameter with an FQDN.

The number of servers maintained in the HA cluster is configured with the **max connections** parameter. Thus if the **max connections** parameter is set to 3, the Net-Net SBC maintains 1 active external policy server with 2 back up servers.

Standby Server Prioritization

The Net-Net SBC looks at the priority and weight fields in the DNS response to create the preferred order of the primary, secondary, tertiary, and quaternary Policy Servers. These 4 addresses are known as the top-level PSs in the cluster.

The Net-Net SBC uses the priority and weights according to RFC 2782. However, weights only apply when multiple servers share the same priority value. If the priorities are different, then the Net-Net SBC will not use weights. If the priorities are the same, then weight of the two (or more) contested servers is used to determine which one to use.

Server States

An HA cluster can contain up to 4 Policy Servers, where the TCP/Diameter connection is established and monitored. Diameter session traffic is only sent to the active policy server in the cluster. The policy servers exist in the following states:

- active-TCP and Diameter connection established. Net-net SBC using this server for policy decisions. The policy server with the highest priority/weight begins in this state.
- standby- TCP and Diameter connection established. Server in standby mode.
- inactive - Diameter connection not successfully established. Net-Net SBC tries to reconnect to inactive servers.

HA Cluster Refresh

The Net-Net SBC sends follow-up SRV queries to the DNS server to refresh the list of available policy servers in the cluster in the following instances:

- DNS cache expires after the TTL is exceeded
- a new policy server with FQDN for an address is configured, saved, and activated on the Net-Net SBC
- after an SPU switchover, the newly active SPU performs a DNS query

When the Net-Net SBC re-queries the DNS server for Diameter external policy servers, the cluster is refreshed with the new/changed servers. Policy server priority is also reconfigured based on newly returned priorities and weights. Upon a cluster refresh, the Net-Net SBC:

- closes connections with standby policy servers that are no longer in the cluster
- creates connections with policy servers which are new in the set

If the currently active policy server remains a member of the cluster after a refresh, it remains active even if its priority has changed. If the active-before-the-refresh policy server is not a member of the cluster after a DNS refresh, the Net-Net SBC gracefully closes the connection to this server. The Net-Net SBC then installs the highest priority server as the active policy server

DNS Failure

If the Net-Net SBC fails to get a response from the DNS server or does not receive at least one IP address in the SRV RR, it continues to send SRV queries periodically, starting with 5 seconds and doubling the interval for every sequential failure, until it receives a valid response. While waiting for a successful DNS response, the Net-Net SBC uses the existing Diameter servers in the DNS cache.

Policy Server Failover

The active external policy server fails over to the highest priority standby server when:

- the TCP connection is closed due to a RST or FIN
- the Diameter connection (CER/CEA exchange) could not be established
- a configured number of consecutive Diameter message timeouts occur. This number is configured in the **max timeouts** parameter. The max timeouts parameter refers to all Diameter messages except for the following:
- You can configure Diameter keepalive message time-outs separately from all other Diameter messages by setting **watchdog ka timer** in the external policy server configuration element. This will failover the active policy server based on a timeout value for DWR messages.
- You can configure Diameter STR message time-outs separately from all other Diameter messages by setting the **str-retry=<timeout number>** option in the external policy server configuration element. This will failover the active policy server based on a unique timeout value for STR messages.

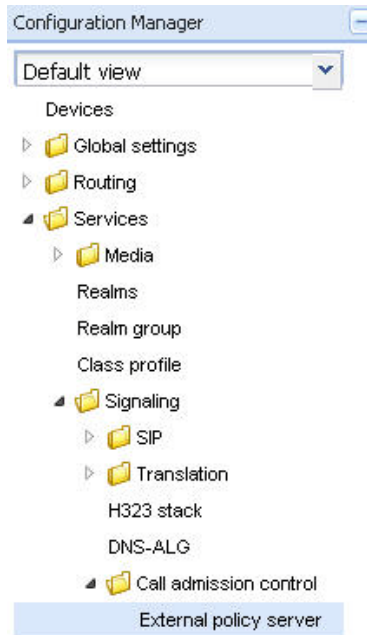
If the Net-Net SBC sends an AAR/STR message to the active policy server and then switches to a different policy server, any new Diameter messages related to that session are sent to the same policy server as long as it is not inactive. If that server becomes inactive, messages will be sent to the new policy server. The new policy server however will not recognize the sessionID and discard the request.

Configuring the External Bandwidth Manager

To configure the external bandwidth manager:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.

- Click External policy server.



- IP address**—Enter the IP address or FQDN of the external policy server. To use external policy server redundancy, you must enter the address as a FQDN.
- Port number**—If you configure this parameter, it will override the port returned in a DNS reply.

IP address	172.30.80.28
Port number	80

- Maximum number of external policy servers to be monitored**—Enter the number of servers to maintain in an external policy server cluster. Valid values range from 1 - 20.

Maximum number of external policy servers to be monitored	1

- Server selection strategy**—Leave this parameter at its default, *Failover* setting.

Server selection strategy	Failover

- Maximum number of reequest timeouts**—Enter the number of request timeouts before the Net-Net SBC sets this external policy server to inactive. You can separately set the number of DWR timeouts that trigger a server to be inactive as an option. Valid values range from 0 - 200.

Maximum number of request timeouts	0

- Click **Apply**.

CLF/Diameter e2 Error Handling

When the Net-Net SBC receives an error, as defined in RFC 3588, in a UDA message from a CLF, it replies to the UA with a 503 Service Unavailable message. There are

exceptions for certain values embedded in either the Result-Code AVP or Experimental Result-Code AVP.

Result-Code AVP

If the CLF sends a Result-Code AVP in a UDA message with a value of `DIAMETER_UNABLE_TO_COMPLY` (5012), the Net-Net SBC forwards the REGISTER message using the default configured location string in the P-Access-Network-Info header. If the configured default string is left blank the Net-Net SBC forwards the SIP REGISTER message without the P-Access-Network-Info header.

Experimental-Result-Code AVP

If the CLF sends a grouped Experimental Result AVP in a UDA message that is not successful, as per RFC 3588, the Net-Net SBC performs the following actions with respect to receipt of these two result code values:

- `DIAMETER_ERROR_USER_UNKNOWN`(5001): The Net-Net SBC inserts the configured default-string in for the P-Access-Network-Info header when forwarding the SIP message.
- `DIAMETER_USER_DATA_NOT_AVAILABLE` (4100): The Net-Net SBC inserts the configured default-string in for the P-Access-Network-Info header when forwarding the SIP message.

For both cases, if the configured default string is left blank the Net-Net SBC forwards the SIP REGISTER message without the P-Access-Network-Info header. For all other failure result codes, the Register is rejected.

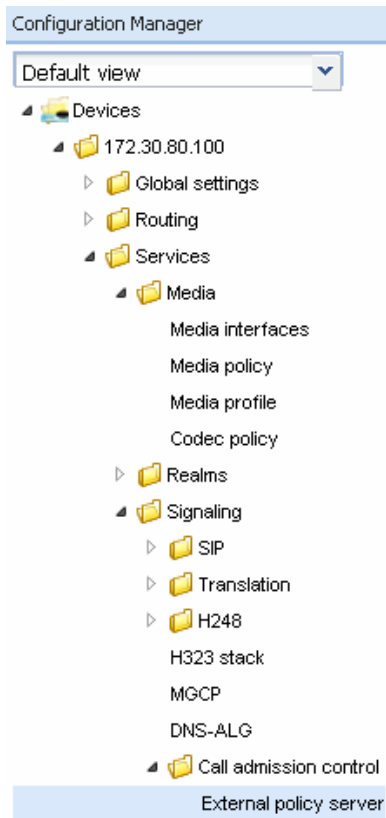
Diameter CLF/e2 Error Bypass

When a CLF returns error messages i.e., any other Diameter answer than `DIAMETER_SUCCESS`, you can configure the Net-Net DD to override the default behavior and still forward REGISTER messages with P-Asserted-Network-Id header containing the **Default location string** parameter value. If the **Default location string** parameter value is empty, the REGISTER message is forwarded without the P-Access-Network-Info header. This error bypass feature is configured by setting the **Permit on reject** parameter to **enabled** for an external policy server configuration element with application-mode set to e2.

Configuration Instructions

To configure the external policy server for use with a CLF:

1. Expand the Configuration Manager slider > Services > Signaling > Call admission control.

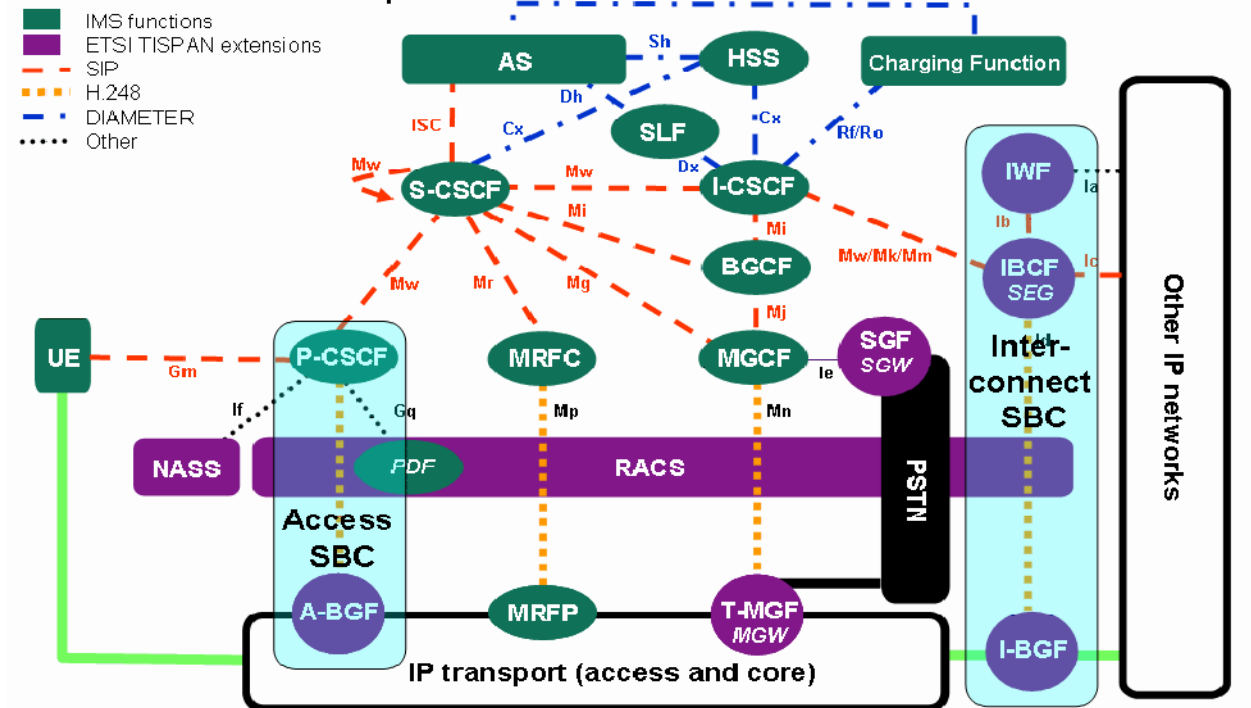


2. Click External policy server.
3. Select an External policy server from the table and click **Edit**.
4. **Permit on reject**—Change this parameter to enabled if you want the Net-Net SBC to forward the session on at a “best-effort”. Leave this parameter set to disabled (default), if you want the Net-Net SBC to deny the session on attempts to revert to the previously-requested bandwidth. (The default value is disabled.)
5. **Default location string**—Set this parameter to the default location string you want inserted into a P-Access-Network-Info header for when the CLF server does not return a unique location string.
6. Click **Apply**.

Introduction

The ETSI TISPAN NGN defines several subsystems that make up the NGN architecture. The model for the target NGN architecture is depicted below. Acme Packet's Net-Net Session Director is an integrated session control, policy enforcement and media management solution that incorporates functional components of the IP multimedia subsystem (IMS) the Resource and Admission Control Subsystem (RACS) and functions necessary for the interconnection with other IP networks/domains. The functions of the Net-Net SBC within the NGN architecture are divided into the interconnect border functions and the access border functions. The diagram below depicts the mapping of these functions across IMS architecture.

Source: TISPAN-3GPP Workshop



Net-Net SBC Access Border Functions

- Proxy CSCF (P-CSCF)
- Access/Core Border Gateway Function (A/C-BGF)
- RACF AF and SPDF functions

Net-Net SBC Interconnect Border Functions

- Interconnect Border Control Function (I-BCF)
- Interworking Function (IWF)
- Interconnect Border Gateway Function (I-BGF)

IMS Access Border Functions

The Net-Net SBC is deployed as the access point between the core IMS network and UEs to deliver the functions defined in the TISpan architecture as the P-CSCF, and A-BGF. These two functions can not be separated.

P-CSCF Functions

The Net-Net SBC performs the following functions in the role of P-CSCF:

- Forwards SIP REGISTER messages and maintains a cached mapping of the user info and the UE's Address of Record (AoR), including the far-end NAT address in the case of hosted NAT traversal (HNT).
- Forwards SIP messages to a S-CSCF based on service route discovery procedures.
- Performs local emergency session handling—Local routing policy is used by the Net-Net SBC to identify emergency sessions and provide unique routing (e.g. can route to a dedicated S-CSCF function for emergency session handling).
- Operates as a UA (B2BUA) for generating independent SIP transactions for security purposes and handling of abnormal conditions.
- Offers current session timers which are used to monitor for media faults and abandoned calls.
- Generation of CDRs—The Net-Net SBC generates real-time accounting records via RADIUS.
- Authorization of bearer resources and QoS management—With integrated BGF capabilities, the Net-Net SBC allocates bearer resources (NAPT flows) and applies QoS policies (including packet marking) based on local policies and/or policies acquired via interaction with the A-RACF (PDF).
- Interaction with the A-RACF (PDF) for session-based policy enforcement and admission control—The Net-Net SBC PDF interface options include COPS and SOAP/XML.
- Traffic Policing—Traffic is policed at the session and media/transport layer. At the signaling layer, the Net-Net SBC polices at a number of levels including:
 - Capacity—Total number of concurrent calls to/from each realm
 - Session set-up rate—Maximum rate of call attempts to/from each signaling element
 - Signaling message rate—Each endpoint's signaling message rate is monitored and policed
 - Signaling bandwidth—each endpoint's signaling bandwidth is policed individually

A-BGF Functions

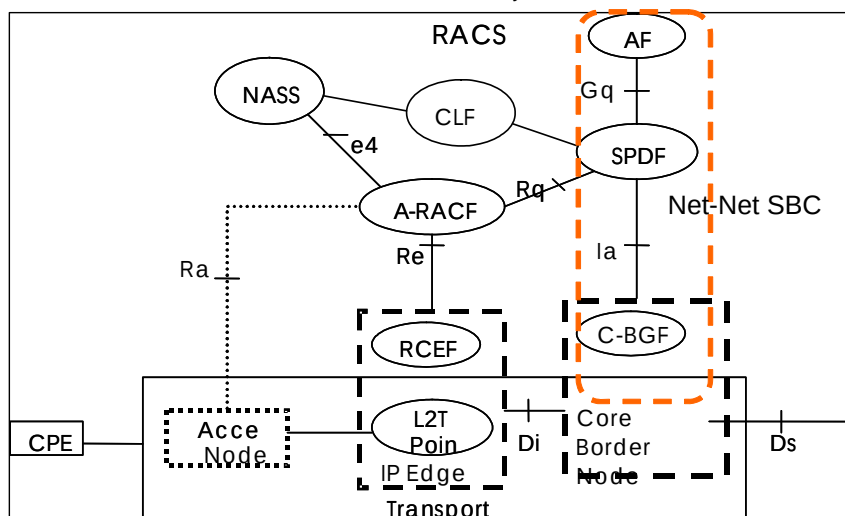
The Net-Net SBC performs the following IMS BGF functions:

- Opening and closing gates/packet filtering—The Net-Net SBC opens and closes gates (media pinholes) on a session-by-session basis. Packet filtering rules include full source and destination IP address and port number.
- Per-session DiffServ or ToS marking—Media flows destined for the IMS core network can be explicitly marked using ToS or DiffServ. Media packets can be marked by VPN, by codec (voice, video) or by E.164 phone number prefix.

- **NAPT-PT and topology hiding**—The Net-Net SBC provides NAPT for all media flows associated with a session on a per session-basis. Double NATing, NATing both source and destination sides, is utilized to fully hide topology in each direction for RTP and RTCP. Local IP addresses and port resources are dynamically allocated from steering pools provisioned on the Net-Net SBC.
- **Hosted NAT traversal**—The Net-Net SBC supports HNT function that allows media flow traversal through the CPE firewall/NAT without upgrading the CPE equipment. The Net-Net SBC interacts with the endpoints to dynamically establish and maintain bindings in the CPE firewall/NAT that allow the signaled communications to pass through. The Net-Net SBC's registration management and media relay functions make CPE-based NATs transparent to the service delivery elements.
- **Traffic Policing**—Traffic is policed at the session and media/transport layer. At the signaling layer, the Net-Net SBC polices at a number of levels including policing of Media (e.g. RTP & RTCP) traffic on a per-flow basis—CBR policing is applied to each flow based on negotiated offered and negotiated media codecs.

Resource and Admission Control (RACS) Functions

The figure below illustrates the mapping of Net-Net SBC functions to the RACS functional model. In this model, the Net-Net SBC incorporates the Application Function (in the case of IMS this is the P-CSCF function), the SPDF (Service Policy Decision Function) and the Core Border Gateway function.



The Net-Net SBC, acting as the SPDF, interfaces with the PDF (A-RACF policy decision function) for resource authorization and admission control on a call-by-call basis. COPS is the supported PDF interface.

IMS Interconnect Border Functions

The Net-Net SBC is deployed at IP interconnect points between service providers to deliver the functions defined in the TISpan architecture as the I-BCF, IWF and I-BGF.

Interworking Function (IWF)

- Interworking SIP profiles and other protocols (e.g. H.323)

Interconnect Border Control Function (I-BCF)

- Interaction with I-BGF (including NAPT and firewall functions)
- Insertion of the IWF when appropriate
- Topology hiding—screening of signalling information

Interconnect-Border Gateway Function (I-BGF)

- Gate opening/closing
- NAPT and packet filtering
- Packet marking
- Resource allocation and bandwidth reservation
- Security and topology hiding
- Session admission control, resource and traffic management
- Upstream/downstream flow policing
- Quality monitoring and reporting
- Usage metering - CDR generation
- Lawful Intercept

IMS Path and Service Route Header Support

The Net-Net SBC supports the Path header and the Service-Route header used in the registration phase of a SIP transaction. The Net-Net SBC will learn the route vectors from the SIP URIs contained in these headers in order to preload SIP headers with the correct route vectors in subsequent SIP message exchanges between the UA and the S-CSCF across the Net-Net SBC. This is how the Net-Net SBC supports RFC 3608 and RFC 3327.

Path Header

When a UE registers to an S-CSCF, the Net-Net SBC adds the Path header in the REGISTER message as it is proxied to the S-CSCF. The Path header includes the SIP URIs that form the route vector which describes how the UE reaches the Net-Net SBC, through a specific series of proxies. This route vector is saved in the Net-Net SBC's registration entry for the UE, routing all subsequent SIP messages from the S-CSCF to the UE. As the Path header is sent to the S-CSCF, the Net-Net SBC, as P-CSCF, inserts the SIP URI of itself as the top entry in the Path header.

The Path header only appears in SIP messages exchanged during the registration process.

If the REGISTER request already contains a Path header, the Net-Net SBC stores the contents of the Path header(s) for this endpoint for routing back to the endpoint in subsequent messages.

Service Route Header

When a UE registers through the Net-Net SBC to the registrar, the registrar returns a Service-Route header in a 200 OK message in response to the REGISTER message to the UE. This header contains the route vector that directs traffic through a specific sequence of proxies used to reach the S-CSCF. The Service-Route header only appears during the SIP registration process.

The P-CSCF (Net-Net SBC) will now store the URIs listed in the Service-Route header(s) in the registration entry of the UE for use in routing subsequent traffic to the S-CSCF. The Net-Net SBC inserts this sequence of proxies into and outgoing message's Route headers; this is called a pre-loaded route. This route is only applicable for the traffic flowing between the originating UE and the contacted S-CSCF.

When receiving subsequent requests from the UE, the Net-Net SBC looks at the UE's registration entry for a service route, and will insert the route vector as appropriate Route headers. If the service route is not found in the registration entry, the routing is performed in the usual fashion.

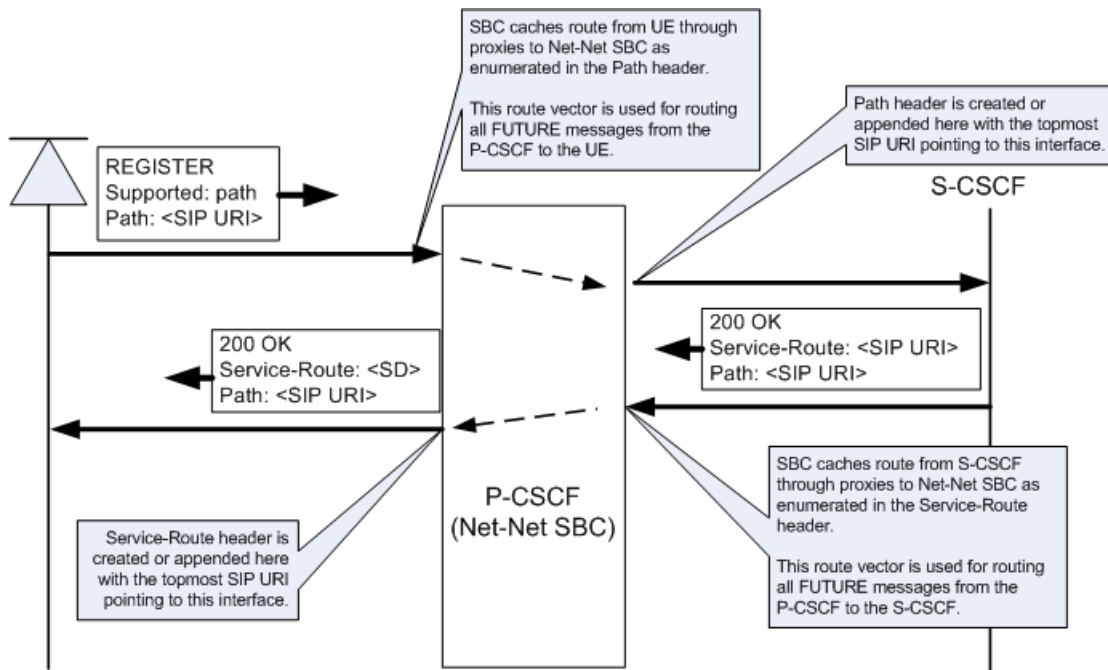
As an exception, you may wish for the Net-Net SBC to not use the Service-Route header to route subsequent Register requests. Note in the configuration section the way to disable Service-Route header routing.

The manner in which passing or stripping of Service-Route headers sent from the S-CSCF is done is determined by local configuration on the Net-Net SBC. There is no verification of configured local policy against the route included in the stored service route. The Service-Route header, as created by the Net-Net SBC, and exiting back to the UE, contains a SIP URI pointing to itself as the topmost entry. This is used so that other proxies can learn the route back to the Net-Net SBC.

Summary

If a request originates at the UE, the routes enumerated in the Service-Route header are used to route the request to the S-CSCF. If a request is meant to terminate at a UE, the routes enumerated in the Path header are used to route the response to the UE. Service-Route routes take priority over configured local policy.

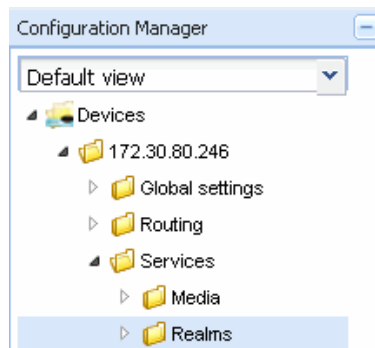
Path headers received in a 200 OK response from the registrar are transmitted to the UE unchanged. If you want them stripped as the SIP message leaves the Net-Net SBC, you can use the SIP Header Manipulation function.



Configuring IMS Path and Service Route Header Support

This section describes how to configure IMS support on your Net-Net SBC from the Net-Net Central.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.

5. Select the SIP realm service for which you want to configure IMS and click **Edit**.
6. **Enable IP multi-media subsystem**—Select enabled from the combination box to enable IMS processing on this SIP interface.

Enable IP multi-media subsystem

7. Click **Apply**.

IMS Support for Private Header Extensions for 3GPP

As part of its RFC 3455 support, the Net-Net SBC supports the following headers in its IMS implementation:

- P-Associated-URI
- P-Asserted-Identity
- P-Called-Party-ID
- P-Charging-Function-Address
- P-Visited-Network-ID

The procedure to enable IMS support is explained under ACLI Configurations and Instructions in the previous section. IMS and all related functions must be enabled on both the access-side and core-side SIP interfaces.

P-Associated-URI Header

In the SIP registration process, the registrar often returns a set of associated URIs for a registering AoR. When the Net-Net SBC receives the list of associated URIs, it stores them in the registration entry for the registering endpoint. The service provider allocates one or more associated URIs per user for his or her own usage. After an endpoint successfully registers, the P-Associated-URI header returned in a 200 OK message informs the UE of all URIs associated with the AoR.

When the Net-Net SBC receives a request from a UE, the URI in the From header is matched against the registration cache for that endpoint. If the registering endpoint matches an associated-URI already in the registration table, the Service-Route associated with this endpoint is used to create the route for originating transactions associated with the endpoint to the S-CSCF.

The inclusion or exclusion of the P-Associated-URI header is not dependent on the trust level of an ingress or egress realm.

P-Asserted-Identity Header

The Net-Net SBC inserts a P-Asserted-Identity header into any initial request for a dialog or standalone transaction sourced by the UE.

The inclusion or exclusion of the P-Asserted-Identity header is dependent on the trust level of an egress realm.

P-Asserted-Identity Header Handling

1. The Net-Net SBC inserts a P-Asserted-Identity header into all messages other than the REGISTER message.
2. When the P-Preferred-Identity header is present in an INVITE sourced by the UE, and the SIP URI contained in this header is also present in the UE's associated URI list, then this SIP URI is inserted in the P-Asserted-Identity header as the SIP message enters the core network.

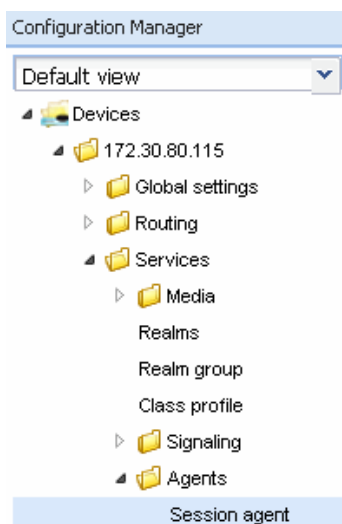
3. When the P-Asserted-Identity header is present in an INVITE sourced by the UE, and the SIP URI contained in this header is also present in the UE's associated URI list, then the original P-Asserted-Identity header and SIP URI is passed unchanged into the core network.
4. When the From header is present in an INVITE sourced by the UE, and the SIP URI contained in this header appears in the UE's Associated URI list, then this SIP URI is inserted into the P-Asserted-Identity header as the SIP message enters the core network.
5. When the P-Asserted-Identity header is present in an INVITE sourced by the UE, and the SIP URI contained in this header is not present in the Associated URI list, the Net-Net SBC acts like no P-Asserted-Identity was received from the UE.
6. When no P-Asserted-Identity can be derived from an INVITE sourced by the UE, the P-Asserted-Identity is based on the first URI in the Associated URI list.
7. The P-Asserted-Identity header will be removed from SIP messages sent and received from a UE if either the ingress or egress side is untrusted and the UE's Privacy header's contents is "id".
8. If no P-Associated-URI exists for a registered endpoint, the Net-Net SBC will use the configured default P-Asserted-Identity found on the sourcing session agent. This feature works with both SIP and H.323 session agents.
9. If the session agent that originates a message does not include a P-Asserted-Identity header or the request is not originated from the session agent, and the P-CSCF has not received P-Associated-URI list from the registrar for a particular user, no P-Asserted-Identity will be created.
10. The P-Preferred-Identity header will never be passed to the S-CSCF.

If the above steps fail to insert a P-Asserted-Identity header, you can manually configure a value to be inserted into a P-Asserted-Identity header. The sip-ims-feature parameter must still be enabled to use the P-Asserted-Identity header override.

Configuring the P-Asserted-Identity Header

This section describes how to configure P-Asserted-Identity header support on your Net-Net SBC from the Net-Net Central. Before performing the following procedures, you must enable the IP multi-media subsystem checkbox.

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Select the SIP session agent for which you want to configure a P-Asserted-Identity header and click **Edit**.
4. **P-Asserted-identity for IMS**—Enter the P-Asserted-Identity header value in the text field.
5. Click **Apply** to continue.

P-Asserted-identity for IMS

sip:name@acmepacket.com

P-Called-Party-ID Header

The Net-Net SBC transparently passes the P-Called-Party-ID header between the S-CSCF and a UA.

IMS Charging Headers

The Net-Net SBC supports IMS Charging Headers. These headers include P-Charging-Vector and the P-Charging-Function-Address. IMS charging header support is configured separately from other IMS functions in order to support a variety of customer needs. Charging header information is now recorded in the CDR records.

A charging vector is defined as a collection of the charging information defined in RFC 3455. It is used to correlate charging records among network elements. The charging vector is constructed during the establishment of the dialog or a standalone transaction outside of a dialog.

Charging headers are inserted, deleted, or ignored for request messages. They are forwarded through the Net-Net SBC unmodified when embedded in response messages. If you wish to modify the charging headers in a response message, you must use the Net-Net SBC's header manipulation feature as a general solution.

P-Charging-Vector

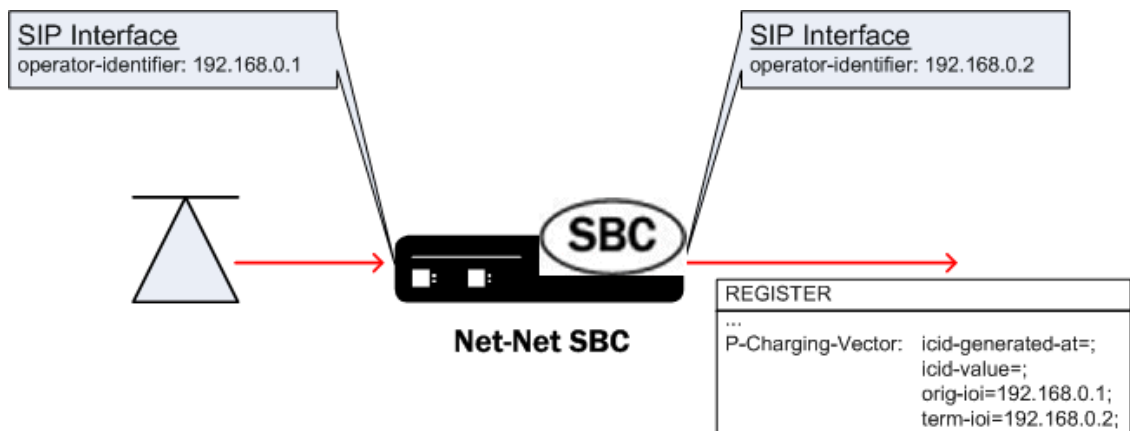
You can configure the Net-Net SBC to process the P-Charging Vector header in three different ways.

- If a P-Charging-vector header is present in an incoming SIP request, the Net-Net SBC can pass the header untouched, as part of the full SIP message that is forwarded out of an egress interface.
- If a P-Charging-vector header is present in an incoming SIP request, the Net-Net SBC can delete the header and forward the full SIP message out of an egress interface.
- If an incoming SIP request does not contain a P-charging-vector header, the Net-Net SBC can create and insert the header and forward the full SIP message out of an egress interface. Likewise, if an incoming SIP request contains an existing P-Charging-Vector header, the Net-Net SBC can overwrite this header with the values generated internally.

The P-Charging-Vector header is composed of four parameters: icid-value, icid-gen-addr, orig-ioi, term-ioi. See RFC 3455, Section 4.6 for more information.

1. The Net-Net SBC constructs the icid-value in the following format: string2@string1 where:
String 1 is the IP address of the egress SIP interface.
String 2 is a unique string value created by the Net-Net SBC and based on the realm, local IP port, time, and a sequence number.
2. The icid-gen-addr parameter's value is the IP address of the egress SIP interface. This value is generated by the Net-Net SBC.
3. The orig-ioi parameter's value is set manually using the operator-identifier field located in the SIP interface configuration element.
4. The term-ioi parameter's value is set manually using the operator-identifier field located in the SIP interface configuration element.

You configure charging vector handling on the Net-Net SBC interface that receives the SIP request by turning on the switches that enable charging vector processing on the ingress interface for the call. Based on the direction of the call, the Net-Net SBC will insert the operator-identifier configuration parameter into the orig-ioi and the term-ioi parameters. The orig-ioi parameter takes the value of the operator-identifier configuration parameter of the SIP interface that receives the SIP request. The term-ioi parameter takes the value of the operator-identifier configuration parameter of the SIP interface that sends the SIP request to its next hop.



P-Charging-Vector Header Example

P-Charging-Vector: icid-value=late6g46n1823s8719ck3ps6gbt46m5d3bci3po5hhdg3n86g1csio47g9c43@192.168.0.2;

```

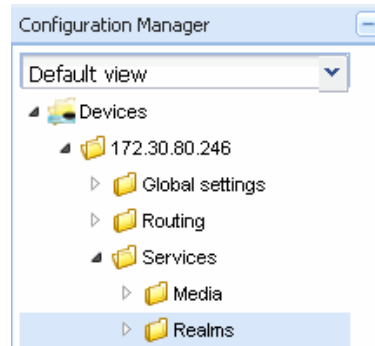
icid-generated-at=192.168.0.2;
orig-ioi=192.168.0.1;
term-ioi=192.168.0.2;

```

Configuring P-Charging-Vector Processing for SIP Interfaces

This section explains how to configure P-Charging-Vector processing using the Net-Net Central.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select the SIP realm service for which you want to configure P-Charging-Vector support and click **Edit**.
6. **Action to be performed on P-Charging-Vector header**—Choose insert from the combination box.
7. **Action to be performed on P-Charging-Function-Address header**—Choose insert from the combination box.
8. **Charging collection function (CCF) address**—Enter the CCF string in the text field.
9. **Event charging function (ECF) address**—Enter the ECF string in the text field.
10. **Operator ID**—Enter the operator identifier string in the text field. This string must begin with an alpha character.

Charging function

Action to be performed on P-Charging-Vector header	insert
Action to perform on P-Charging-Function-Address header	insert
Charging collection function (CCF) address	192.168.0.20
Event charging function (ECF) address	192.168.0.21
Operator ID	ioi-192.168.0.2

11. Click **Apply** to continue.

P-Charging-Function-Address

The P-Charging-Function-Address header is composed of two configurable parameters: ccf, ecf. You can configure the Net-Net SBC to process the P-Charging-Function-Address header in the following ways.

- If a P-Charging-Function-Address header is present in an incoming SIP request, the Net-Net SBC can be set to pass the header, untouched, as the full SIP request is forwarded out of an egress interface.
- If a P-Charging-Function-Address header is present in an incoming SIP request, the Net-Net SBC can be set to delete the header and forward the SIP request out of an egress interface.
- If an incoming SIP request does not contain a P-Charging-Function-Address header, the Net-Net SBC can be set to create and insert the header and forward the SIP message out of an egress interface.
- If an incoming SIP request contains a P-Charging-Function-Address header, and the Net-Net SBC is set to insert a configured P-Charging-Function-Address header, the new parameters will be appended before the existing parameters in the header. The Net-Net SBC will then forward the SIP request out of an egress interface.

P-Charging-Function-Address Header Example

P-Charging-Function-Address: ccf=192.168.0.20 ; ecf=192.168.0.21;

SIP Charging Function Address Enhancements for IMS

The Net-Net SBC's handling for the 3GPP P-Charging-Function-Address (PCFA) header has been augmented to allow this header, found in REGISTER responses and dialog-creating or stand-alone messages, to be cached.

There are two values you can set for the **Action to be performed on P-Charging-Function-Address header** parameter in the ingress SIP interface configuration:

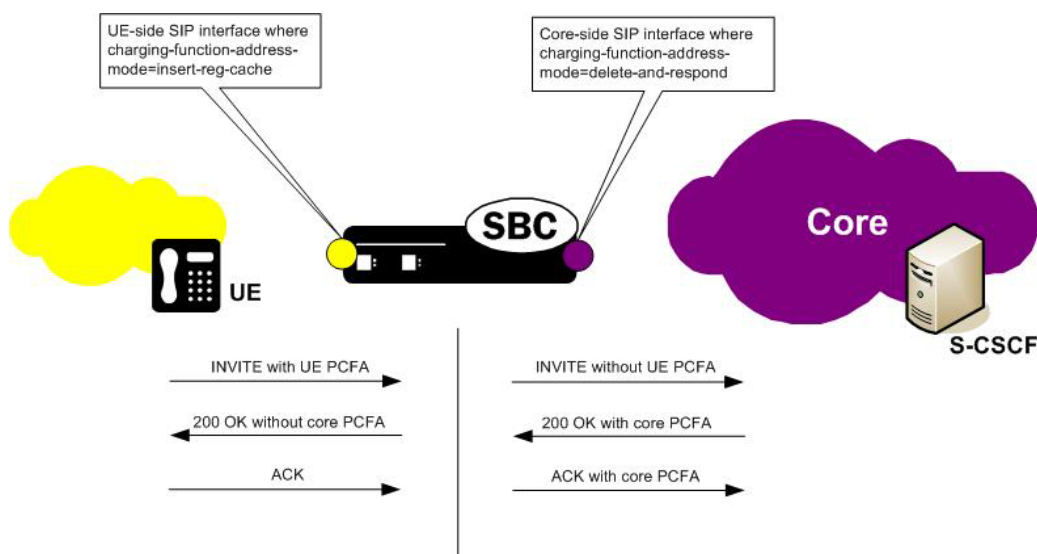
- **insert-reg-cache**—To be configured on the SIP interface facing the UE, this value configures the Net-Net SBC to replace the PCFA with the most recently cached values rather than the **Charging collection function (CCF) address** and **Event charging function (ECF) address** you set to be static in your configuration. The cached values come from one of the following that the Net-Net SBC has received most recently:
 - Request and response messages for the same dialog
 - Registration cache entry
 - Local configuration (statically configured in **Charging collection function (CCF) address** and **Event charging function (ECF) address** parameters)
- **delete-and-respond**—To be configured on the SIP interface facing the S-CPCF, this value configures the Net-Net SBC to strip out the latest cached PCFA, and then cache the latest PCFA from responses.

Notes on Prior Behavior

In order to introduce these two new modes, the Net-Net SBC's prior behavior has been modified slightly. Formerly, the PCFA header was counted in RADIUS CDRs for all responses; however, since responses can come from a UE, this behavior can result in updating accounting information with untrusted headers. The Net-Net SBC no longer behaves in this way. In addition, accounting record information can now be updated during a session for PCFA functions.

How It Works

This section describes how PCFA insertion and caching work on the Net-Net SBC. The following diagram shows how this feature can be configured to strip PCFAs out of messages from UEs, and to use core PCFAs in communications with the core.



PCFA Insertion

Using the two new settings, the Net-Net SBC adds the PCFA to all SIP requests and to the response on the S-CPCF facing the SIP interface. However, only dialog-creating and standalone requests, and responses to each of those, update the Net-Net SBC and accounting information. Such requests do not have a To tag, and responses do not appear in established dialogs. The Net-Net SBC inserts the PCFA into provisional (1XX) and success (2XX) responses, with the exception of the **100 Trying** response.

You can use SIP header manipulation rules (HMR) to remove any unwanted headers.

PCFA Caching

When you use either of the new configuration parameters, the Net-Net SBC uses the latest cached copy of a PCFA header to insert into requests and responses. The Net-Net SBC does not cache any PCFA headers it receives on SIP interfaces using the none, insert, or insert-reg-cache modes because this type of SIP interface faces the UE making its replacement headers ones from the core.

Though there can be various sources for the latest cached copy, the PCFA header received as part of a dialog-creating or standalone request has highest precedence. This PCFA header is then stored as the latest cached value for that dialog. That is, for each specific dialog, the Net-Net SBC the PCFA is cached separately so it can add the most specific PCFA to the message—and is added to any message for the dialog.

When there is no cache PCFA for a specific dialog, the Net-Net SBC uses the registration cache entry as the latest cached copy. And when there is no entry in the registration, the PCFA uses the **CCF address** and **ECF address** values from the SIP interface.

The latest cached copy or the **CCF address** is the value reported in the RADIUS VSA Acme-Session-Charging-Function-Address; this VSA is used for both of the new modes. Note that only the **CCF address** is reported in RADIUS records; the **ECF address** is not.

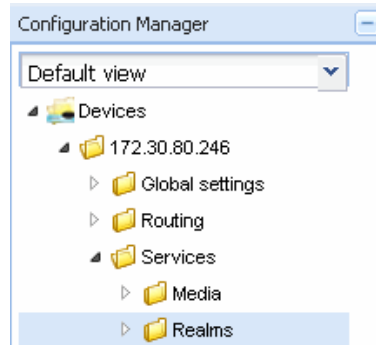
Configuring SIP Charging Function Address Enhancements

For this feature to work as designed, you need to configure:

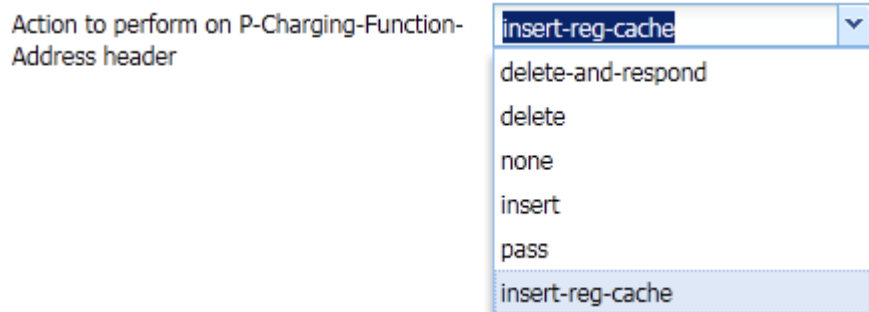
- The UE-facing SIP interface with a **Action to be performed on P-Charging-Function-Address header** value of **insert-reg-cache**
- The core-facing SIP interface with a **Action to be performed on P-Charging-Function-Address header** value of **delete-and-respond**

To set the charging function address mode:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select the SIP realm service to which you want to apply SIP route header removal in the table.
6. Click **Edit**.
7. **Action to be performed on P-Charging-Function-Address header**—Choose the charging function address mode you want to use from the combination box.



- **none**—Pass the P-Charging-Vector header received in an incoming SIP message untouched as the message is forwarded out of the Net-Net SBC, but does not include icid-value in accounting records
- **pass**—Pass the P-Charging-Vector header received in an incoming SIP message untouched as the message is forwarded out of the Net-Net SBC, includes icid-value in accounting records
- **delete**—Delete the P-Charging-Vector header received in an incoming SIP message before it is forwarded out of the Net-Net SBC
- **insert**—Inserts the P-Charging-Vector header in an incoming SIP message that does not contain the P-Charging-Vector header. If the incoming message contains the P-Charging-Vector header, the Net-Net SBC will

overwrite the P-Charging-Vector header with its values. This option always uses the **CCF address** and **ECF address** static values.

- **insert-reg-cache**—To be configured on the SIP interface facing the UE, configures the Net-Net SBC to replace the PCFA with the most recently cached value rather than the **CCF address** and **ECF address** you set to be static in your configuration. The cached values come from one of the following that the Net-Net SBC has received most recently: request, response, registration, or local configuration.
- **delete-and-respond**—To be configured on the SIP interface facing the S-CPCF, configures the Net-Net SBC to strip out the latest cached PCFA from the core side. The Net-Net SBC then remembers this PCFA and uses it in communications sent to the core.

Note that the default settings for this parameter and for **Action to be performed on P-Charging-Vector heading** are **pass** for new SIP interface configurations. If you are upgrading and there are pre-existing SIP interfaces in your configuration, the defaults become **none**.

8. Click **Apply**.

RADIUS Accounting of Charging Headers

When the Net-Net SBC creates a either the P-Charging-Vector header or the P-Charging-Function-Address header, it inserts an entry in the RADIUS record to record the charging header data.

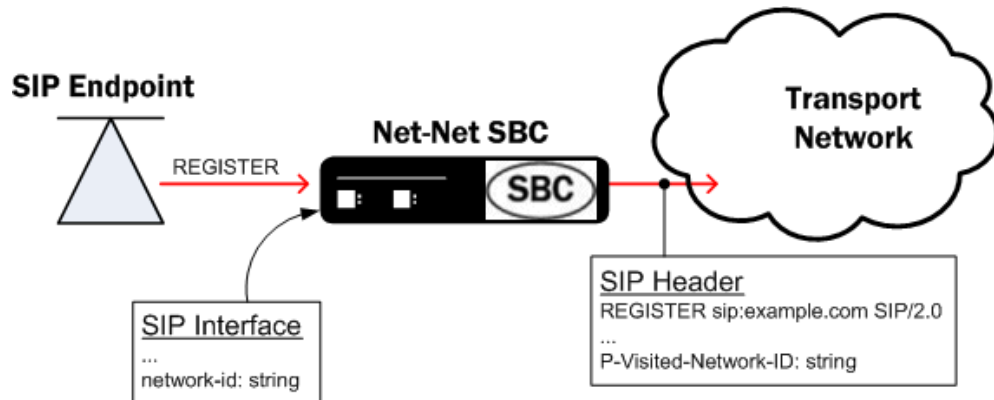
For a P-Charging-Vector header, the icid-value is saved to the P-Charging-Vector attribute in the radius record. If the Net-Net SBC does not create a P-Charging-Vector header, but it receives a SIP message that already has the P-Charging-Vector header with an icid-value, the existing icid-value is written to the RADIUS record.

For a P-Charging-Function-Address header, the first CCF value is saved to the P-Charging-Function-Address attribute. When the Net-Net SBC creates the P-Charging-Function-Address, the CCF value it inserts into the header is saved to the radius record. If the Net-Net SBC does not create a P-Charging-Function-Address header, but it receives a SIP message that already has the P-Charging-Function-Address with a CCF value, the existing CCF value is written to the RADIUS record.

Name	Value	Value Type
Acme-Session-Charging-Vector	54	string
Acme-Session-Charging-Function-Address	55	string

P-Visited-Network-ID Header

The Net-Net SBC's IMS support also includes the insertion of a P-Visited-Network-ID header into SIP messages when applicable. When a UE sends a dialog-initiating request (e.g., REGISTER or INVITE message) or a standalone request outside of a dialog (e.g., OPTIONS) to the P-CSCF, the Net-Net SBC inserts the P-Visited-Network-ID header into the SIP message as it enters into the destination network.

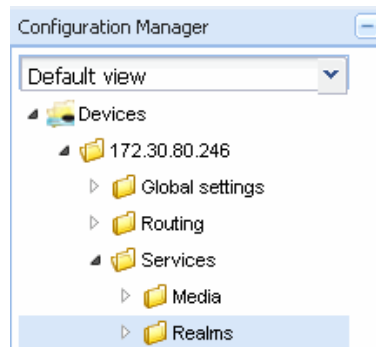


The P-Visited-Network ID header will be stripped from SIP messages forwarded into untrusted networks as expected. The content of a P-Visited-Network-ID header is a text string that identifies the originating UE's home network. This string is user-configurable.

Configuring P-Visited-Network-ID Header Handling for SIP Interfaces

This section explains how to configure P-Visited-Network-ID header handling using the Net-Net Central. Before performing the following procedures, you must enable the IP multi-media subsystem checkbox.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.

5. Select the SIP realm service for which you want to configure IMS and click **Edit**.
6. **Enable IP multi-media subsystem**—Select enabled from the combination box to enable IMS processing on this SIP interface.

Enable IP multi-media subsystem enabled ▼

7. Click **Apply**.

SIP IMS and Privacy

In order to determine the trust level on the ingress side correctly, you need to set the **ppi-to-pai** parameter in the SIP interface configuration. If the sip-ims-feature parameter is enabled on the incoming SIP interface, then the Net-Net SBC applies the IMS rules for determining PAI. And if the PAI is based on the IMS processing, the PPI will be removed. Otherwise, the Net-Net SBC removes the PPI unless you:

- Set the ingress SIP interface's **ppi-to-pai** parameter to the replace mode
- Set the ingress SIP interface is designated as trusted and **ppi-to-pai** is also set to trusted

The egress processing of this parameter is applied if the PPI is not removed.

On the egress SIP interface, the Net-Net SBC checks the ppi-to-pai parameter only if the message's destination is designated trusted. If the destination is untrusted, then the Net-Net SBC does not reference this parameter.

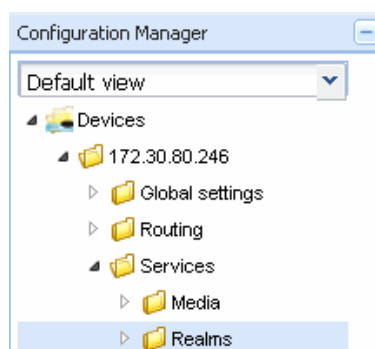
There are four valid values for the **ppi-to-pai** parameter:

- **none**—The Net-Net SBC performs no conversion
- **trusted**—The Net-Net SBC checks the configured trust level for the ingress endpoint; if it is untrusted, then PPI headers in the message are deleted; if it is trusted, then PPIs are converted to PAIs
- **remove**—The Net-Net SBC removes PPI headers
- **always**—The Net-Net SBC always converts PPIs to PAIs

Setting the PPI to PAI Conversion Mode

This section explains how to set the PPI to PAI conversion mode.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the SIP realm for which you want to configure PPI to PAI conversion and click **Edit**.

4. Select the Services/Protocols tab.
5. Select the SIP realm service you want to configure and click **Edit**.
6. **Preferred/Asserted identification conversion mode**—Select the conversion mode you want to use from the drop-down list.
7. Click **Apply**.

Recurse 305-Only

The Net-Net SBC has a SIP feature called redirect action. This is a feature that allows the Net-Net SBC, acting as a SIP Proxy or a Session Agent, to redirect SIP messages after receiving a SIP redirect (3xx) response. Previously, for the parameters of SIP interface and session agent on the Net-Net SBC, you could set the the redirect-action parameter to “**proxy**” or “**recurse**”. In Release 6.3 you can additionally set a value of “**recurse-305-only**” for the redirect-action parameter.

Redirect Action Process

When the redirect-action parameter is set to “**proxy**” (which is the default), the Net-Net SBC sends SIP Redirect responses back to the previous hop (back to the User Agent Client (UAC)) when the User Agent Server (UAS) is not a session agent. The URI in the Contact of the response is changed from the URI that was in the original request.

Note: If the target of the request is a session agent with redirect action defined, its redirect action by default is “recurse” which overrides the SIP interface’s redirect action.

When the redirect-action parameter is set to “**recurse**”, if the Net-Net SBC receives a SIP redirect (3xx) response on the SIP interface, it automatically redirects all requests to the Contact URI specified in the 3xx response. The responses contain the same Contact URI that was in the original request sent to the UAS.

For example, if UAC X sends an INVITE to the Net-Net SBC set up as a SIP proxy, the Net-Net SBC forwards the INVITE to UAS Y (Y is not a session agent). Y then responds to the Net-Net SBC with a 3xx response (redirection message) with the same URI that was in the original request. This indicates to the Net-Net SBC that if it receives any future requests directed toward Y, that it should automatically redirect the request directly to Y. The Net-Net SBC then recurses, or repeatedly sends subsequent incoming messages to the Contact URI specified in the Header of the 3xx responses.

When the redirect-action parameter is set to “**recurse-305-only**”, if the Net-Net SBC receives a 305 SIP redirect response (Use Proxy) on the SIP interface, it automatically redirects all requests to the Contact URI specified in the 305 response. All other 3xx responses are sent back to the previous hop.

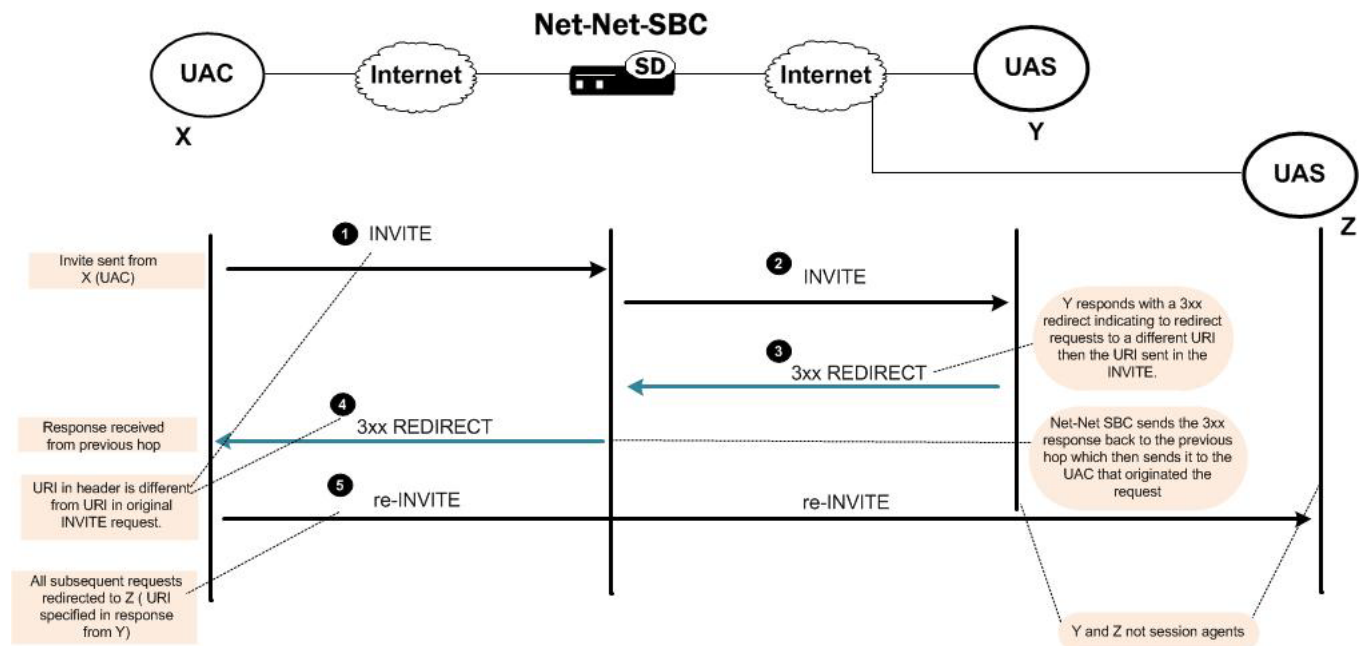
When the UAS is a session agent, the Net-Net SBC can send the SIP redirect response back to the UAC using the value in the session agent’s redirect action field. If there are too many UASs to define as individual session agents, or if the UASs are Hosted NAT Traversal (HNT) endpoints, and SIP redirect responses need to be proxied for UASs that are not session agents, you can set the default behavior at the SIP interface level.

Redirect-Action Set to “Proxy”

The following occurs if you set the **redirect-action** parameter to “proxy” on the Net-Net SBC:

8. X (UAC) sends an INVITE to the SD.
9. The SD forwards the INVITE to Y (UAS).
10. Y sends the 3xx REDIRECT response to the SD with a different URI in the message header.
11. The SD forwards the 3xx REDIRECT response to the previous hop.
X receives the 3xx REDIRECT response from the previous hop.
12. X redirects all subsequent requests to the URI in the message header received from Y.

The following illustration shows an example of a dialog between X, Y, Z, and the SD during a redirect-action session set to “proxy”.

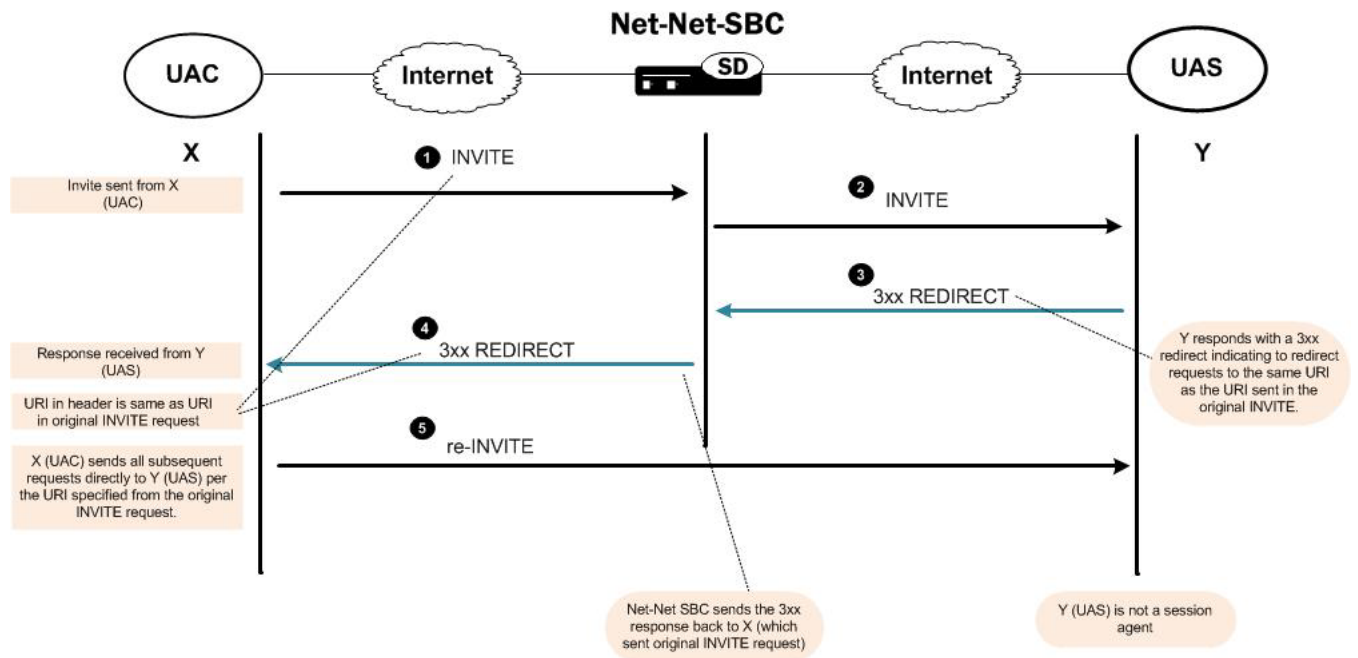


Redirect-Action Set to “Recurse”

The following occurs if you set the **redirect-action** parameter to “recurse” on the Net-Net SBC:

1. X (UAC) sends an INVITE to the SD.
2. The SD forwards the INVITE to Y (UAS).
3. Y sends the 3xx REDIRECT response to the SD with the same URI as the URI sent in the original request.
4. The SD forwards the 3xx REDIRECT response to X (UAC).
5. X (UAC) sends all subsequent requests directly to Y (UAS) per the URI specified from the original INVITE request.

The following illustration shows an example of a dialog between X, Y, and the SD during a redirect-action session set to “recurse”.

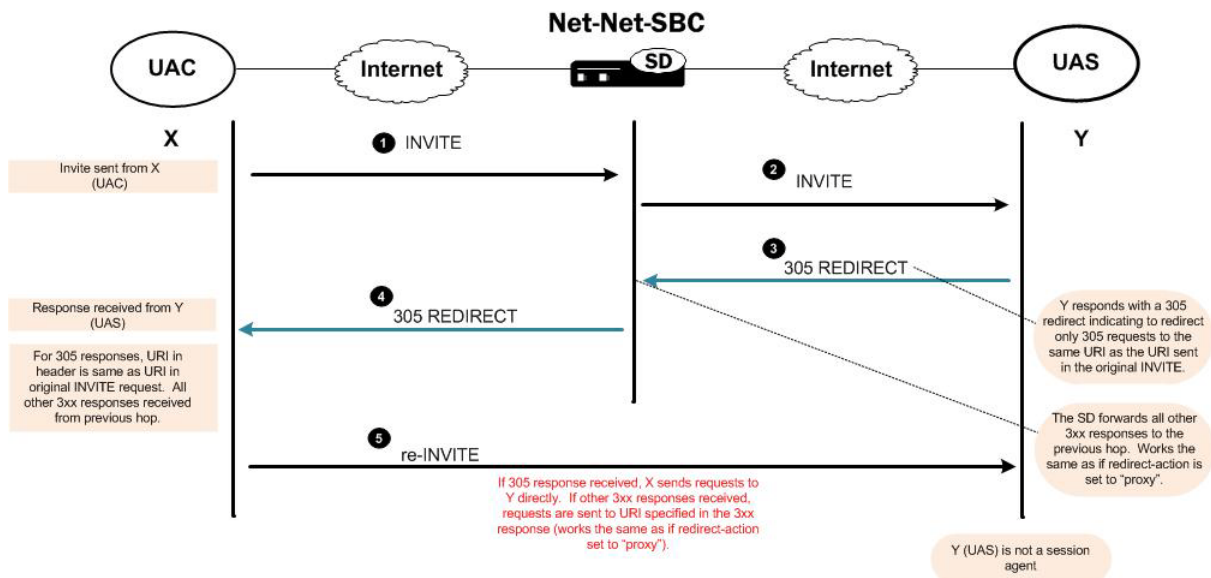


Redirect-Action Set to “Recurse-305-Only”

The following occurs if you set the **redirect-action** parameter to “recurse-305-only” on the Net-Net SBC:

1. X (UAC) sends an INVITE to the SD.
2. The SD forwards the INVITE to Y (UAS).
3. Y sends a 305 REDIRECT response to the SD with the same URI as the URI sent in the original request.
4. The SD forwards the 305 REDIRECT response to X (UAC).
5. If 305 response received, X sends requests to Y directly. If other 3xx responses received, requests are sent to URI specified in the 3xx response (works the same as if redirect-action set to “proxy”).

The following illustration shows an example of a dialog between X, Y, and the SD during a redirect-action session set to “recurse-305-only”.



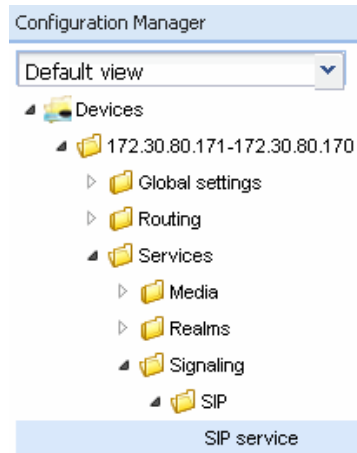
Configuring “Recurse 305 Only”

You can configure the Net-Net SBC to redirect requests from a UAC to a UAS using the URI in 305 responses only. You can configure this feature on the SIP interface or for a session agent.

SIP Interface

To configure the redirect-action feature on the SIP interface on the Net-Net SBC:

1. Expand the Configuration Manager slider > Services > Signaling - > SIP.



2. Click SIP service.
3. Click the SIP service you want to configure in the table and click **Edit**.
4. Click the General folder and scroll to Modes.
5. **Redirect**—Click **Recurse-305-only** in the drop down list. Select the action the SIP proxy takes when it receives a Redirect (3xx) response from the sip interface. If the response comes from a session agent and this field is empty, the redirect action value will be recurse. .



- Values
- **proxy**—SIP proxy passes the response back to the previous hop. The response will be sent based on the proxy-mode of the original request.
 - **recurse**—SIP proxy sends the original request to the list of contacts in the Contact header of the response, serially (in the order in which the contacts are listed in the response)
 - **recurse-305-only**—SIP proxy sends the original request to the list of contacts in the Contact header of only the 305 responses, serially (in the order in which the contacts are listed in the response). All other 3xx responses are sent back to the previous hop.

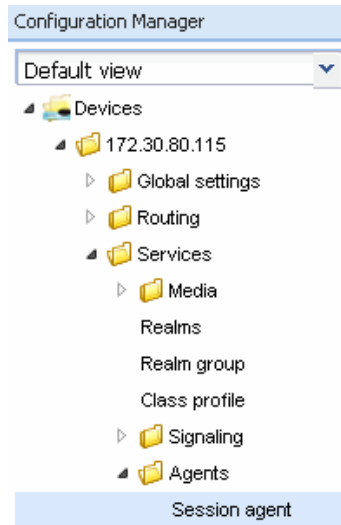
When the Net-Net SBC receives a 305 SIP redirect response (Use Proxy) on the SIP interface, it automatically redirects all requests to the Contact URI specified in the 305 response. All 3xx responses other than 305 responses are sent back to the previous hop.

6. Click **Apply**.

Session Agent

To configure the redirect-action feature for a session agent on the Net-Net SBC:

1. Expand the Configuration Manager slider > Services > Agents.



2. Click Session agent. The Session agent table appears in the content area.
3. Click the session agent you want to configure in the table and click **Edit**.
4. Click the Session agent tab and scroll to SIP message handling.
5. **Action upon receipt of REDIRECT from SA**—Click Recurse-305-only in the drop down list. Select the action the SIP proxy takes when it receives a Redirect (3xx) response from the session agent. If the response comes from a session agent and this field is empty, the redirect action value will be recurse..



- Values
- **proxy**—SIP proxy passes the response back to the previous hop. The response will be sent based on the proxy-mode of the original request.
 - **recurse**—SIP proxy sends the original request to the list of contacts in the Contact header of the response, serially (in the order in which the contacts are listed in the response)
 - **recurse-305-only**—SIP proxy sends the original request to the list of contacts in the Contact header of only the 305 responses, serially (in the order in which the contacts are listed in the response). All other 3xx responses are sent back to the previous hop.

If the target of the request is a session agent with redirect action defined, its redirect action by default is “recurse” which overrides the SIP interface’s redirect action.

When the Net-Net SBC receives a 305 SIP redirect response (Use Proxy) on the SIP interface, it automatically redirects all requests to the Contact URI specified in the 305 response. All 3xx responses other than 305 responses are sent back to the previous hop.

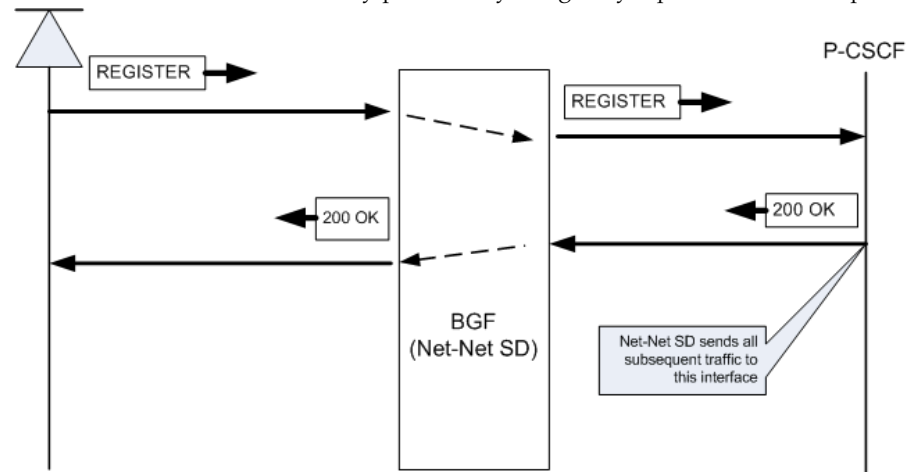
6. Click **Apply**.

Implicit Service Route

In an IMS architecture, the Service-Route header indicates the network's requirement for where to direct all subsequent signaling from a given UE. Upon receipt of the Service-Route header(s) in a 200 OK response to a REGISTER request, the Net-Net SBC stores the values contained therein and preloads them when issuing new requests (such as an INVITE or a SUBSCRIBE) from the associated endpoint.

Although not all SIP networks fully support RFC 3608, some still require UEs to send all call traffic to the network element(s) with which they originally registered. Further, some P-CSCFs do not pass back the Service-Route headers in response to the UE's REGISTER request. This situation requires that a logical connection is made between the UE and the network element to which it registers.

In these scenarios, the Net-Net SBC may create an implicit Service-Route for the success (2xx class) response it receives from the P-CSCF. An implicit Service-Route is useful for creating a binding between a UE and a next-hop proxy, even when the Net-Net SBC can route to many potential, yet logically equivalent next-hops.

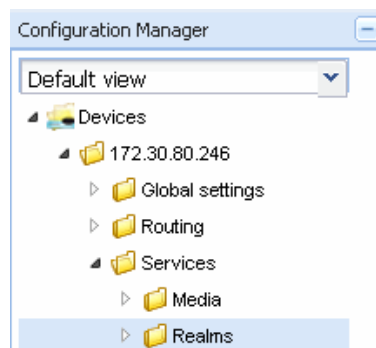


The Net-Net SBC's implicit Service-Route feature is enabled with the implicit-service-route parameter. It is configured on the access-side SIP interface where a UE registers.

Enabling Implicit Service-Route Support

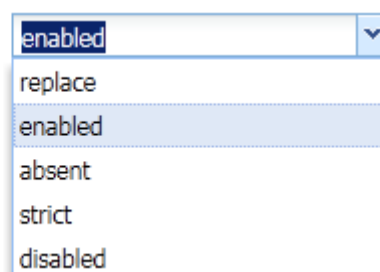
This section explains how to enable implicit service routing on your Net-Net Central.

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select the SIP realm service you want to configure and click **Edit**.
6. Scroll to Implicit service routing.
7. **Implicit service routing**—Select enabled from the combination box. The following are valid values for this parameter:
 - enabled—Implicit service routing is enabled.
 - disabled—Implicit service routing is disabled.
 - strict—Implicit service routing is enabled, but the loose route parameter will not be inserted in the Route URI of the implicit service route for requests sent to the proxy.
 - absent—An implicit service route to the session agent to which the REGISTER request was sent is constructed when the successful REGISTER response contains no Service-Route headers.
 - replace—An implicit service route to the session agent to which the REGISTER request was sent is used to construct the service route. The Net-Net SBC ignores the Service-Route headers in successful REGISTER responses.

Implicit service routing



8. Click **Apply**.

Service Route Use

Once an endpoint registers successfully, the Net-Net SBC caches the Service-Route header (if any) to use for routing all subsequent requests from the endpoint—with the exception of any subsequent REGISTER requests.

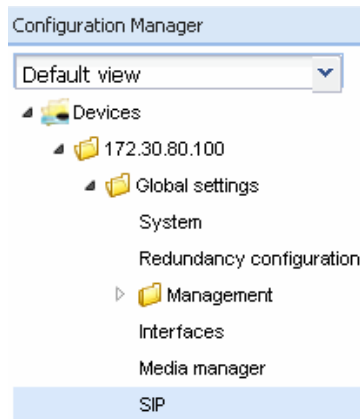
You can set whether or not you want the Net-Net SBC to route subsequent REGISTER requests using the cached Service Route, and whether the endpoint is engaged in an active session through the Net-Net SBC. If you decide not to use the Service Route for endpoints engaged in active sessions, then the Net-Net SBC uses the local policy to make routing decisions.

For endpoints not in found in the Net-Net SBC's registration cache, the Net-Net SBC again uses the local policy to make routing decisions.

Registering Service Routes

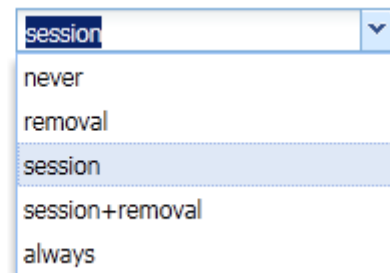
To configure how you want the Service Route used:

1. Expand the Configuration Manager slider > Global settings.



2. Click SIP.
3. Click **Add** to add the SIP configuration in the content area.
4. Click the Registration tab at the top of the content area.
5. **Register service route**—Enter the way you want the Net-Net SBC to use the service route:
 - **always**—The Net-Net SBC always uses the cached service route for an endpoint for routing REGISTER requests.
 - **never**—The Net-Net SBC never uses the service route, and makes routing decisions based on local policies instead.
 - **removal**—The Net-Net SBC uses the cached service route for an endpoint when routing REGISTER requests that remove the endpoint's contact. It uses the local policy for refresh and query REGISTER requests.
 - **session**—The Net-Net SBC uses the cached service route when routing REGISTER requests that appear while an endpoint has an active session traversing it. When an endpoint does not have an active session, the Net-Net SBC uses the local policy to make routing decisions.
 - **session+removal**—Combining the session and removal values, the Net-Net SBC uses the cached service route: when routing REGISTER requests that remove the endpoint's contact and when REGISTER requests appear while an endpoint has an active session traversing the system. Otherwise, the Net-Net SBC uses the local policy to make routing decisions.

Register service route



Click **Apply**.

6. **Network ID**—Enter the network ID string in the text field.

Network ID

examplenetwork-id

7. Click **Apply**.

Surrogate Registration

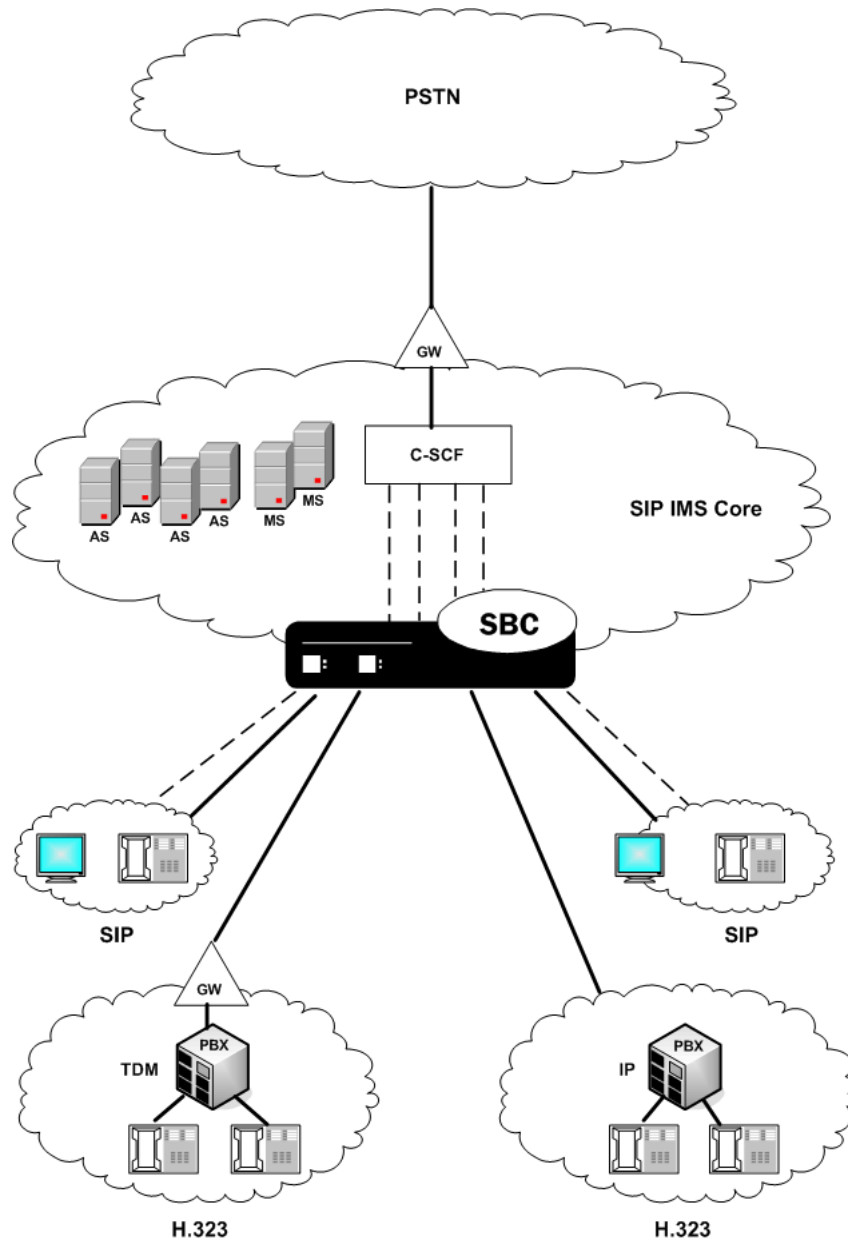
The Net-Net SBC surrogate registration feature lets the Net-Net SBC explicitly register on behalf of a Internet Protocol Private Branch Exchange (IP-PBX). After you configure a surrogate agent, the Net-Net SBC periodically generates a REGISTER request and authenticates itself using a locally configured username and password, with the Net-Net SBC as the contact address. Surrogate registration also manages the routing of class from the IP-PBX to the core and from the core to the IP-PBX.

Integrating with IMS

With surrogate registration, the Net-Net SBC lets IP-PBXes integrate with the IP Multimedia Subsystem (IMS) architecture. The IP-PBX registers itself as if it were user equipment (UE), which triggers the implicit registration of all phone numbers associated with the IP-PBX.

Implicit registration means the explicit registration of one address of record (AoR) triggers the implicit registration of all the other AoRs associated with that UE. The implicitly registered AoRs are passed back to the UE as P-Associated-URIs in the registration's 200 (OK).

IMS assumes that each SIP endpoint can register itself with its Serving-CSCF (S-CSCF). However, phones can be connected to SIP Integrated Access Devices (IADs) or SIP or H.323 IP-PBXes. The Net-Net SBC performs SIP registration on behalf of the IP-PBX and IADs.



How it Works

The Net-Net SBC registers on behalf of the IP-PBXes and then stores the associated URIs returned by the Serving Call Session Control Function (S-CSCF). The calls from the phones behind the IP-PBX can be routed based on the cache entry the Net-Net SBC creates after it receives each phone's associated URI. Calls are routed using the service route, local policy or any other routing mechanism based on the associated session agent or session agent group. The Net-Net SBC also supports multiple registrations on behalf of a IP-PBX because the IP-PBX can support

thousands of phones, but the registrar might only be able to send 10 to 20 associated URIs in response to a single registration.

The Net-Net SBC replaces the Contact URI for requests from the IP-PBX to the core to match the registered value. For calls from the IMS core to the IP-PBX, the Net-Net SBC replaces the Request-URI username with P-Called-Party-ID/To-URI username. The IMS cores sends INVITES for the phones behind the IP-PBX with the registered Contact URI as the Request-URI instead of the AoR of the phones. The IP-PBX needs to see the phone's AoR in the Request-URI.

Registration

The Net-Net SBC uses the configuration information of the surrogate agent that corresponds to a specific IP-PBX. After the surrogate agents are loaded, the Net-Net SBC starts sending the REGISTER requests on their behalf. (You can configure how many requests are sent.)

If the Net-Net SBC receives 401 or 407 responses to REGISTER, requests, it will use the Message Digest algorithm 5 (MD5) digest authentication to generate the authentication information. You need to specify the password. The Net-Net SBC also supports authentication challenge responses with the quality of protection code set to auth (qop=auth), by supporting the client nonce (cnonce) and nonce count parameters.

The Net-Net SBC creates a registration cache entry for each of the AoRs for which it is sending the REGISTER requests. When the Net-Net SBC receives the associated URIs, it checks whether the customer host parameter is configured. If it is configured, the Net-Net SBC changes the host in the received Associated-URI to the customer host. If it is not configured, the Net-Net SBC does not change the Associated-URI. It makes the registration cache entries that correspond to each of the Associated-URIs. The From header in the INVITE for calls coming from the IP-PBX should have one of the Associated-URIs (URI for a specific phone). If the Net-Net SBC receives a Service-Route in the 200 (OK) response, it stores that as well.

The Net-Net SBC uses the expire value configured for the REGISTER requests. When it receives a different expire value in the 200 OK response to the registration, it stores the value and continues sending the REGISTER requests once half the expiry time has elapsed.

REGISTER requests are routed to the registrar based on the configuration. The Net-Net SBC can use the local policy, registrar host and the SIP configuration's registrar port for routing.

If the Net-Net SBC is generating more than one register on behalf of the IP-PBX, the user part of the AoR is incremented by 1 and the register contact-user parameter will also be incremented by 1. For example, if you configure the register-user parameter as caller, the Net-Net SBC uses caller, caller1, caller2 and so on as the AoR user.

Routing Calls from the IMS Core

The calls coming from the core will have the Net-Net SBC's Contact-URI (which is sent in the REGISTER request) as the Request-URI. The Net-Net SBC looks for a registration entry that corresponds to this URI. After finding the registration entry and the corresponding surrogate agent, the Net-Net SBC looks for the routing mechanism it should use to route this INVITE to the IP-PBX. It uses the customer-next-hop configuration parameter to determine if it routes this call to the session agent, the session agent group, or directly to a particular IP address.

SIP

If the customer-next-hop parameter points to a SIP session agent or the SIP session agent group, the Net-Net creates a Route header using the session agent and modifies the Request-URI. It changes the user portion of the Request-URI to either the user portion of the P-Called-Party-ID header, if present, or to the user portion of the To header. The Net-Net SBC also changes the host portion of the Request-URI to the hostname configured in the customer-host configuration parameter. It makes the change because the domain name on the core side can be different than the domain name on the access IP-PBX side. The Net-Net SBC then uses the added Route header to properly route the call.

H.323

If the session agent or the session agent group configured for the customer-next-hop parameter references an H.323 device, the Net-Net SBC sends the INVITE to its interworking task. If a session agent group is being used, the parameter containing the session agent group name is added to the Request-URI. The host portion of the Request-URI will point to the interworking IP address and the port is changed to 1720.

If a session agent is used, the Net-Net SBC uses it to route the call properly to the interworking task to take care of the H.323 call setup.

Routing Calls from the IP-PBX

The Net-Net SBC looks for a match in the registration cache based on the From header or the P-Preferred-Identity header. The header should contain the user portion of one of the Associated-URIs that it received from the registrar in the 200 (OK) responses to REGISTER requests. It should also have the same hostname that is configured in the customer-host parameter. If that parameter is not configured, then the hostname should be same as the one configured for the register-host parameter.

With an H.323 IP-PBX, when the Net-Net SBC receives an INVITE from the interworking task it looks to see if the call is coming from a session agent. If it is, it looks to see if this session agent has a surrogate agent configured. If it does, the Net-Net SBC changes the host portion of the From header to match the registration entry stored in the registration cache.

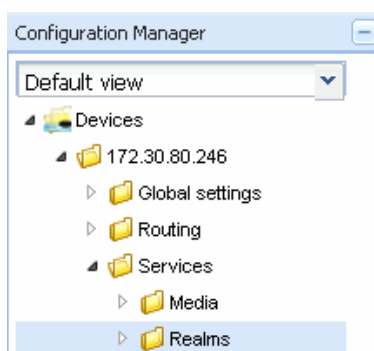
After the corresponding registration Service-Router entry is found, the Net-Net SBC uses the Service-Route for this endpoint to route the call, if it exists. If no Service-Route exists but the SIP interface's route-to-registrar parameter is enabled, the Net-Net SBC tries to route this to the registrar. You can configure the surrogate agent to override the SIP interface's route-to-register setting. If the surrogate agent's route-to-register parameter is set to disable, it takes precedence over the SIP interface's setting. The Net-Net SBC will not try to route the call to the registrar.

Configuring Surrogate Registration

You can configure surrogate registration using EMS. You need to configure a surrogate agent for each IP-PBX proxy for which the Net-Net SBC will be registering. Those parameters that are optional are marked, the rest are mandatory.

To configure the surrogate agent:

1. Expand the Configuration Manager slider > Services.

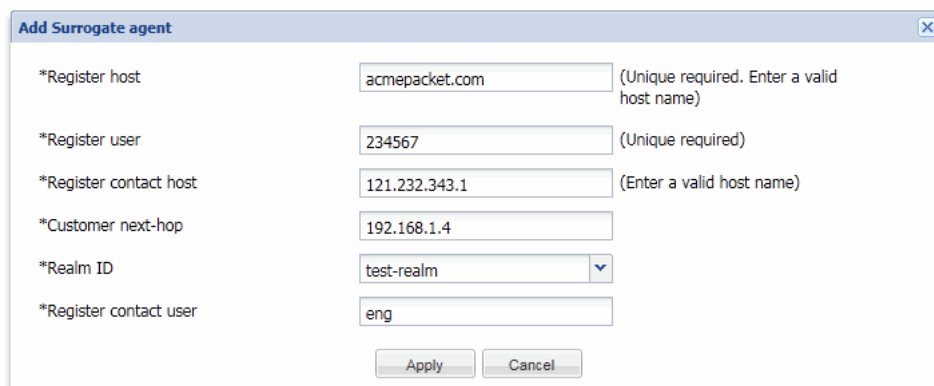


2. Click Realms. The Realm table appears in the content area.
3. Select the realm for which you want to configure a surrogate agent and click **Edit**.
4. Click the Surrogate registration tab at the top of the content area.

Adding a Surrogate Agent**To add a surrogate agent:**

1. Click **Add**. The Add Surrogate agent window appears.
2. **Register host**—Enter the registrar's hostname to be used in the Request-URI of the REGISTER request. This name is also used as the host portion of the AoR To and From headers.
3. **Register user**—Enter the user portion of the AoR.
4. **Customer next-hop**—Enter the next hop to this surrogate agent:
 - session agent group:
SAG: <session agent group name>
 - session agent:
<hostname> or <IPV4>
 - specific IP address:
<IPV4> or <IPV4: port>
5. **Register contact host**—Enter the hostname to be used in the Contact-URI sent in the REGISTER request. This should always point to the Net-Net SBC. If specifying a IP address, use the egress interface's address. If there is a SIP NAT on the registrar's side, use the home proxy address in the SIP NAT

6. **Register contact user**—Enter the user part of the Contact-URI that the Net-Net SBC generates..



The 'Add Surrogate agent' dialog box contains the following fields and values:

Field	Value	Notes
*Register host	acmepacket.com	(Unique required. Enter a valid host name)
*Register user	234567	(Unique required)
*Register contact host	121.232.343.1	(Enter a valid host name)
*Customer next-hop	192.168.1.4	
*Realm ID	test-realm	
*Register contact user	eng	

Buttons: Apply, Cancel

7. Click **Apply** to complete adding this surrogate agent. This surrogate agent appears in the Surrogate agent table.

Editing the Surrogate Agent

To edit the surrogate agent:

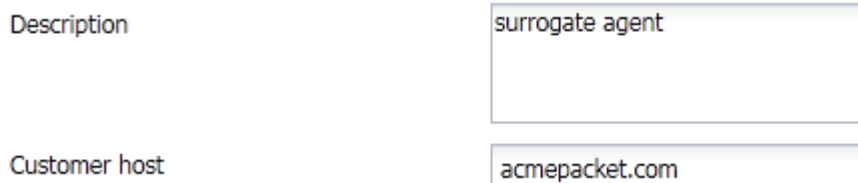
1. Select the surrogate agent you just created and click **Edit**.
2. **State**—Select enabled from the combination box to indicate whether the surrogate agent is used by the application.



The 'State' dropdown menu shows the following options:

- enabled (selected)
- disabled

3. **Description**—*Optional*. Enter a description of this surrogate agent.
4. **Customer host**—*Optional*. Enter the domain or IP address of the IP-PBX, which is used to determine whether it is different than the one used by the registrar.



The 'Description' field contains the text 'surrogate agent'.

The 'Customer host' field contains the text 'acmepacket.com'.

5. **Password**—Enter the password if you are configuring the auth-user parameter. This is in case the registrar sends the 401 or 407 response to the REGISTER request.
6. **Register expires (sec)**—Enter the expires in seconds to be used in the REGISTER requests. Default value is 600,000 (1 week).



The 'Password' field is empty.

The 'Register expires (sec)' field contains the value '600000'.

7. **Replace contact**—Set to enabled or disabled (default). This specifies whether the Net-Net SBC needs to replace the Contact in the requests coming from the

surrogate agent. If this is enabled, Contact will be replaced with the Contact-URI the Net-Net SBC sent in the REGISTER request.

Replace contact	disabled
Route to registrar	enabled disabled

8. **Route to registrar**—Set to enabled (default) or disabled. This indicates whether requests coming from the surrogate agent should be routed to the registrar if they are not explicitly addressed to the Net-Net SBC.

Route to registrar	enabled enabled disabled

9. **AOR count**—Enter the number of registrations to do on behalf of this IP-PBX. If you enter a value greater than 1 (default), the Net-Net SBC increments the register-user and the register-contact-user values by that number. For example, if this count is 3 and register-user is john then users for three different register messages will be john, john1, john2. It does the same for the register-contact-user values.
10. **Auth user**—Enter the authentication user name you want to use for the surrogate agent. This name is used when the Net-Net SBC receives a 401 or 407 response to the REGISTER request and has to send the REGISTER request again with the Authorization or Proxy-Authorization header. The name you enter here is used in the Digest username parameter. If you do not enter a name, the Net-Net SBC uses the value of the register-user parameter.

AOR count	1
Auth user	user

11. *Optional.* Click **Add** to enter non-standard options or features.

Options		Options..

The Option dialog box appears.

12. **Option**—Enter the name of the option you want to use.

User defined option	option1	Add

13. Click **Add**. The option name appears in the Options list.
14. Click **OK** to return to the Surrogate registration table. The new configuration appears in the list.

SIP Surrogate Registration Enhancements

For IMS-E networks, the Net-Net SBC's SIP surrogate registration capabilities enable it to register a series of endpoints on behalf of a set of devices that are unable to register themselves. In addition, the Net-Net SBC retries failed registrations, prevents authentication loops, and sends an SNMP trap for failed retransmissions.

Previous Behavior

In previous releases, the Net-Net SBC's surrogate agent performs a series of registrations based on count when the system boots or when its configuration changes. It only attempts to register each user once. Although the surrogate agent uses the same retry mechanism used for SIP client transactions, it does not attempt further if it receives a failure response until the entry expires. When it receives 401, 403, or 407 responses to requests that include authentication, the surrogate agent's automatic incrementing mechanism appends a number to the end of each registered username. Always starting at one, this number cannot appear in any other position in the username.

New Behavior

The Net-Net SBC also supports:

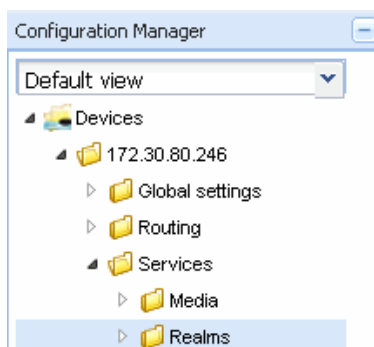
- **Registration retry**—You can configure the surrogate agent to retry registration when after a failure, timeout, or transport error. You can set how many times the Net-Net SBC will attempt to register each user; a setting of zero means retries are unlimited. You can also define the number of seconds to wait before initiating a retry. The Net-Net SBC tracks each registration retry count and timers, and sends an SNMP trap when it reaches the maximum number of retries, which signifies failed registration.
- **Authentication loop prevention**—Authentication loops can occur in previous releases when the Net-Net SBC resends a registration request with authentication in response to 401, 403, or 407 responses (indicating, for example, that there might be a password error). Using the new enhancements, the Net-Net SBC only allows permits the retransmission of one request. It now considers further 401, 403, or 407 responses to be errors and initiates the retry mechanism.
- **Automatic increment enhancements**—The automatic increment works with the caret (^) in the register-user and register-contact-user fields. These carets define where the automatically generated incrementing number is inserted in the username. You can also use multiple carets to define leading zeroes to insert; for example, the entry `user^^^^` will become `user0001`. You can also define the starting integer for the incrementing registrations. For example, setting the AoR count to 20, the count start to 5, and using the value `user^^^^` for register-user and register-contact-user results in the incremented user registrations `user0005` through `user0025`.

Configuring

This section shows you how to configure registration retry mechanism and time, and where to start the count for multiple registrations.

To configure :

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm for which you want to configure a surrogate agent and click **Edit**.
4. Click the Surrogate registration tab at the top of the content area.
5. Click the surrogate agent you want to edit and click **Edit**.
6. **Maximum number of retry attempts**—Enter the number of times you want to attempt registration from 0 (meaning registration attempts are unlimited) to 10.

Maximum number of retry attempts

3

7. **Time between retries (sec)**—Enter the amount of time in seconds, between 30 and 3600 seconds, you want the Net-Net SBC to wait before reattempting registration.

Time between retries (sec)

300

8. **Count start**—Change this parameter from its default of 1 if you want the automatic increment count to start at any other number when the Net-Net SBC performs multiple registrations. The valid range is 0 through 999999999.

Count start

1

SNMP Support

Each time a user fails in attempts to register more times than you have allowed in your configuration (in the **max-register-attempts** parameter), the Net-Net SBC sends the new apSysMgmtSurrogateRegFailed traps, with associated varbinds for the failing AoR and registrar-host. This has been added to `ap-sysmgmt.mib`.

```

apSysMgmtSurrogateRegHostOBJECT-TYPE
SYNTAX          DisplayString (SIZE (1..255))
MAX-ACCESSaccessible-for-notify
STATUS          current
DESCRIPTION
"The registrar host name for a surrogate registration"
::= { apSysMgmtMonitorObjects 41 }

```

```

apSysMgmtSurrogateRegAorOBJECT-TYPE
SYNTAX          DisplayString (SIZE (1..255))
MAX-ACCESSaccessible-for-notify

```

```

STATUS    current
DESCRIPTION
"The address of record used in a surrogate registration"
::= { apSysMgmtMonitorObjects 42 }

apSysMgmtSurrogateRegFailed NOTIFICATION-TYPE
OBJECTS   { apSysMgmtSurrogateRegHost,
             apSysMgmtSurrogateRegAor }
STATUS    current
DESCRIPTION
" The trap will be generated if a surrogate registration
  failed after the maximum configured attempts."
::= { apSystemManagementMonitors 41 }

```

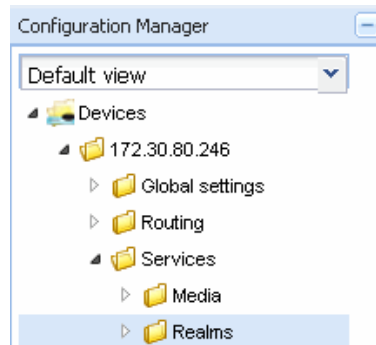
IMS Charging Vector Mode Adaptation

This adaptation to the Net-Net SBC's IMS functionality provides the ability to remove the P-Charging-Vector from incoming requests for a session and store it. Then the Net-Net SBC inserts it into outbound responses related to that session in a P-Charging-Vector header.

Configuring P-Charging-Vector

To configure the P-Charging-Vector on the SIP interface:

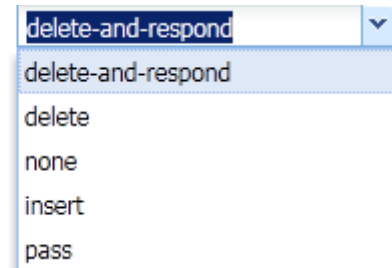
1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the Services and Protocols tab at the top of the content area.
5. Select the SIP realm service for which you want to configure IMS and click **Edit**.
6. Scroll to Charging function.

7. **Action to be performed on P-Charging-Vector header**—Click **delete-and-respond** from the combination box to remove the P-Charging-Vector from incoming requests for a session and store it.

Action to be performed on P-Charging-Vector header



8. Click **Apply**.

IMS: P-CSCF Endpoint Identification Using Address and Port

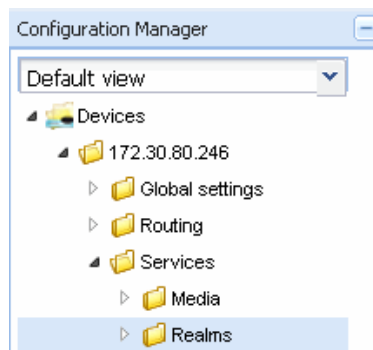
You can configure the Net-Net SBC, acting as a Proxy Call Session Control Function (P-CSCF), to match a request it receives to a registration cache entry based only on the IP address and port from which the request came. When you enable this behavior, the Net-Net SBC will perform this kind of endpoint identification even when nothing in the message matches the cache entry.

Configuring P-CSCF Endpoint Identification

For this behavior to work as designed, you must also have the **reg-via-key** option enabled for the SIP interface to which you are adding the **reg-via-match** option.

To configure P-CSCF endpoint identification using address and port:

1. Expand the Configuration Manager slider > Services.



2. Click **Realms**. The Realm table appears in the content area.
3. Select a SIP realm in the table and click **Edit**.
4. Click the **Services and Protocols** tab at the top of the content area.

5. Select the SIP realm service for which you want to configure IMS and click **Edit**.
6. Scroll to Options.
7. Click **Options**. The Options dialog box appears.
8. Click **reg-via-match** in the Available options list to select it.
9. Click  to move the option to the Selected options list. The option name appears along with a description in the lower part of the window.
10. Click **OK**.
11. Click **Apply**.

IMS-AKA

The Net-Net SBC supports IP Media Subsystem-Authentication and Key Agreement (IMS-AKA).

Defined in 3GPPv7 (specifications in TS 33.203 and call flows in TS 24.228), IMS-AKA can be used as a framework for authentication and for securing the signaling path between a UE and the Net-Net SBC (when the Net-Net SBC is acting as a P-CSCF or as a B2BUA) across the Gm interface.

In addition, the Net-Net SBC's serving as an IMS-AKA termination point is valuable because it allows IMS-AKA use behind by multiple endpoints sitting behind a NAT device. IMS-AKA support also works when there are no NAT devices between endpoints and the Net-Net SBC is acting as a P-CSCF, and when the Net-Net SBC sits behind a third-party P-CSCF. In addition, you can use IMS-AKA when the endpoint uses SIP UDP.

Requirements

IMS-AKA use assumes that you have installed the appropriate IPSec and SSM modules on your Net-Net SBC, or that it has come from Acme Packet with those modules pre-installed. IMS-AKA will not work without this hardware.

In addition, your configuration must have SIP registration caching enabled.

Enabling IMS-AKA

You enable IMS-AKA by configuring the following:

- An IMS-AKA profile
- Certain parameters in the global IPSec configuration
- Certain parameters in the SIP interface, and in the SIP interface's SIP port

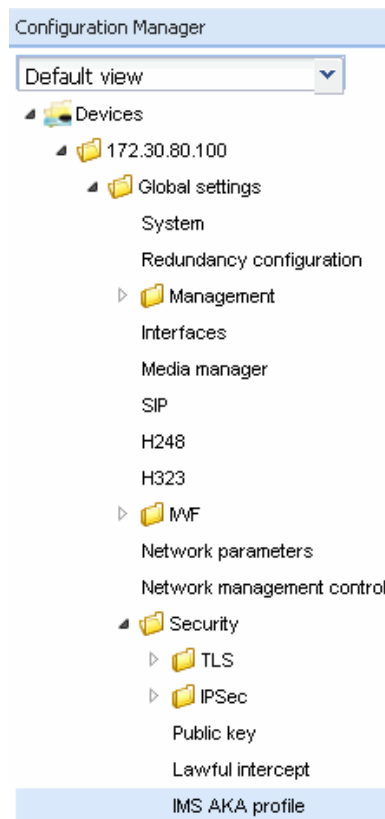
Configuring an IMS-AKA Profile

An IMS-AKA profile establishes the client and server ports to be protected, and it defines lists of encryption and authentication algorithms the profile supports. You can configure multiple IMS-AKA profiles, which are uniquely identified by their names.

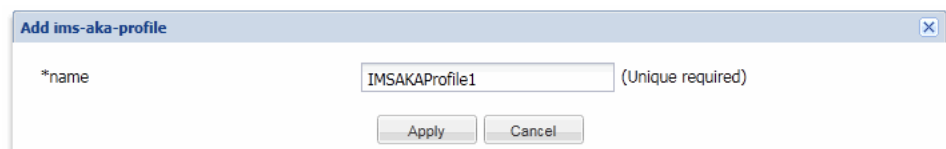
You apply an IMS-AKA profile to a SIP port configuration using the name.

To configure an IMS-AKA profile:

1. Expand the Configuration Manager slider > Global settings > Security.



2. Click IMS AKA profile.
3. Click **Add** to configure a new IMS AKA profile. The Add IMS AKA Profile dialog box appears.
4. **Name**—Enter the name you want to give this IMS-AKA profile. This is the value you will use to apply the profile to a SIP port configuration. This parameter is required, and it has no default value.



5. Click **Apply**. The new IMS AKA profile is added to the table.
6. Click the newly-added IMS AKA profile and click **Edit**.
7. **Protected server port**—Enter the port number of the protected server port, which is the port on which the Net-Net SBC receives protected messages. The protected server port should not overlap with the port range defined in the steering ports configuration using the same IP address and the SIP interface. If there is overlap, the NAT table entry for the steering port used in a call will prevent SIP messages from reaching the system's host processor.

This parameter defaults to 0, which disables the function associated with the parameter. The valid range for values is 1025 to 65535.

protected-server-port

1025

8. **Protected client port**—Enter the port number of the protected client port, which is the port on which the Net-Net SBC sends out protected messages. Like the protected server port, the protected client port should not overlap with the port range defined in the steering ports configuration using the same IP address and the SIP interface. If there is overlap, the NAT table entry for the steering port used in a call will prevent SIP messages from reaching the system's host processor.

This parameter defaults to 0, which disables the function associated with the parameter. The valid range for values is 1025 to 65535.

protected-client-port

1025

9. **encr-alg-list**—Click **Add**. The Add Encryption algorithm list dialog box appears. Click the combination box to choose a value: **aes-cbc** or **des-ede3-cbc**.

This parameter defaults to the following three values: **aes-cbc**, **des-ede3-cbc**, and **null**.

encr-alg-list

10. Click **Apply**.
11. **Authentication algorithm list**—Click **Add**. The Add Authentication algorithm dialog box appears.
12. Click the combination box to choose a value.

This parameter defaults to **hmac-sha-1-96**.

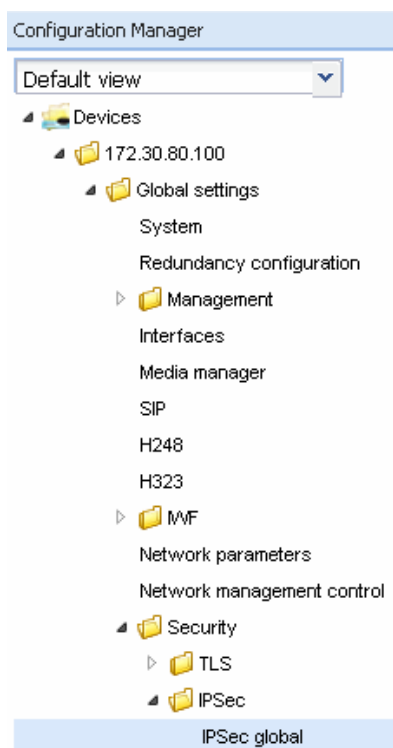
13. Click **Apply**.
14. Click **Apply**. Your settings appear in the IMS AKA profiles table.

Configuring an IPSec Profile for IMS-AKA Use

Using the global IPSec configuration, you establish the parameters governing system-wide IPSec functions and behavior. This configuration also contains parameters required for IMS-AKA support. The IPSec global configuration is a single instance element, meaning there is one for the whole system.

To configure the global IPSec parameters that apply to IMS-AKA:

1. Expand the Configuration Manager slider > Global settings > Security > IPSec.



2. Click IPSec global.
3. **IPsec port to listen for redundancy**—Enter the port on which the Net-Net SBC should listen for redundancy IPSec synchronization messages. The default is 1994, and valid values are in the range from 1025 to 65535.
4. **Maximum number of redundancy sync transactions**—Enter the maximum number of redundancy transactions to retain on the active. The default is 10000, and valid values range up to a 999999999 maximum.
5. **Timeout for checking transaction from standby to active (secs)**—Enter the time in milliseconds before the system starts to send redundancy synchronization requests. The default is 5000, and valid values range up to a 999999999 maximum.
6. **Timeout for subsequent sync requests (secs)**—Enter the time in seconds to define the timeout for subsequent synchronization requests once redundancy synchronization has completed. The default is 1000, and valid values range up to a 999999999 maximum.

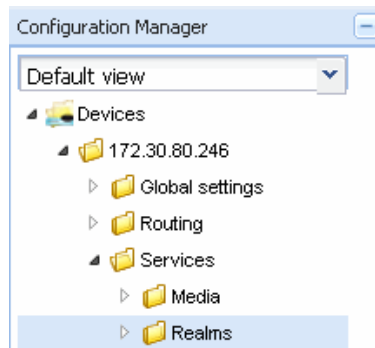
IPSec port to listen for redundancy	1994
Maximum number of redundancy sync transactions	10000
Timeout for checking transaction from standby to active (sec)	5000
Timeout for subsequent sync requests (sec)	1000

- Click **Apply**.

Enabling IMS-AKA Support for a SIP Interface

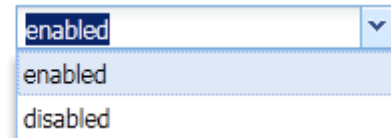
To enable IMS-AKA for a SIP interface:

- Expand the Configuration Manager slider > Services.



- Click Realms. The Realm table appears in the content area.
- Select a SIP realm in the table and click **Edit**.
- Click the Services and Protocols tab at the top of the content area.
- Select the SIP realm service for which you want to enable IMS and click **Edit**.
- Scroll to Enable IMS AKA.
- Enable IMS AKA**—Select enabled from the combination box to enable IMS AKA if you want to use IMS AKA on this SIP interface. By default, this parameter is **disabled**.

Enable IMS AKA

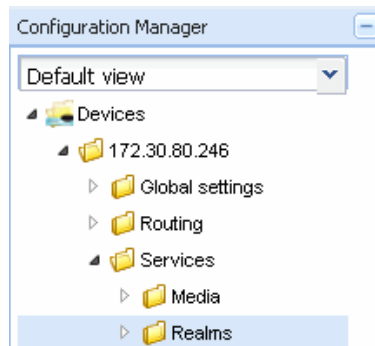


- Click **Apply**.

Applying an IMS-AKA Profile to a SIP Port

To apply an IMS-AKA profile to a SIP port:

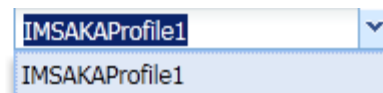
- Expand the Configuration Manager slider > Services.



- Click Realms. The Realm table appears in the content area.
- Select a SIP realm in the table and click **Edit**.
- Click the Services and Protocols tab at the top of the content area.

5. Select the SIP realm service for which you want to apply an IMS profile and click **Edit**.
6. **IMS AKA profile**—Click the name for the IMS-AKA profile configuration you want applied to this SIP port from the combination box. This parameter has no default.

IMS AKA profile


 A screenshot of a web interface showing a dropdown menu. The menu is open, displaying a list of options. The top option, 'IMSAKAProfile1', is highlighted in blue. Below it, another option 'IMSAKAProfile1' is visible. The dropdown is part of a form labeled 'IMS AKA profile'.

7. Click **Apply**.

E-CSCF Support

An Emergency Call Session Control Function (E-CSCF) is an IMS core element that aids in routing emergency calls to an appropriate destination, such as a PSAP. E-CSCF functionality can be performed by the Net-Net SBC with appropriate local policy and network management control configuration.

The E-CSCF feature let the Net-Net SBC internally prioritize and route emergency calls to the corresponding Emergency Service Center, based either on the calling party's request URI, or based on location information retrieved from a CLF (Connectivity Location Function) for wireline/TISPAN networks.

By integrating E-CSCF functionality into the P-CSCF (Net-Net SBC), networks can satisfy the common local requirement that certain telephony elements be deployed locally, rather than use single, centralized elements. Functions like the E-CSCF likely fall into this category.

Service URN Support

To enable E-CSCF functionality, the Net-Net SBC can parse service URNs for local policy lookup keys, and as destination identifiers in network management controls (NMC). Ensure that the match-URN is entered correctly as: "urn:service:sos" or "urn:service:sos.type" or the Net-Net SBC will interpret the URN as a hostname. Please see RFC 5031 for more information on compliant URN construction.

E-CSCF Configuration Architecture

There are four elements which comprise and enable E-CSCF support on the Net-Net SBC:

1. CLF Connectivity
2. NMC Emergency Call Control
3. Local Policy
4. Emergency Local Route Table

CLF Connectivity

The Net-Net SBC must be configured with Diameter-based CLF support. This is accomplished by creating an appropriate external policy server configuration. Please see the *Connectivity Location Function* section in the *External Policy Servers* chapter for more information.

When the Net-Net SBC requests authorization from the CLF server, a Line-Identifier AVP which includes a location string is expected to be returned for the call. The returned location string will be used later for an LRT query.

NMC Emergency Call Control

By configuring a Network Management Control (NMC), the Net-Net SBC can flag a call for special priority early after it is received and validated by the system. The **Destination ID** must be configured in the NMC with the service URN of an incoming emergency call. Also, the NMC configuration must have its **next hop** parameter left blank. This lets the Net-Net SBC route the emergency call with local policies.

For example, if **urn:service:sos** is the configured value in the NMC's **Destination ID**, and an INVITE arrives on the Net-Net SBC with urn:service:sos in the request URI, the call will be flagged for emergency handling. The next step in call processing is for the INVITE to be evaluated by local policy.

Local Policy

Local policies must be configured to match and then route an incoming emergency call. Once a local policy match is made, the Net-Net SBC looks to the configured policy attributes for where to forward the INVITE. A matching policy attribute's next hop should be configured to point to an emergency LRT that contains specific destinations for emergency calls. In addition, the **Emergency location string lookup** parameter must be set to enabled so the Net-Net SBC will perform an LRT lookup based on the location string returned in the CLF response.

The **Emergency LRT lookup key** parameter identifies the attribute, whose value in the location string will be used as the lookup key in the emergency LRT. For example, if the returned location string is:

```
loc=xxx;noc=yyyy;line-code=zzzz
```

and the **Emergency LRT lookup key** parameter is set to **noc**, then when the Net-Net SBC performs a local policy route search, it will search the LRT for yyyy. If the **Emergency LRT lookup key** parameter left empty or if there is no match when **Emergency location string lookup** is enabled, the entire location string is used as the lookup key.

Emergency LRT

The Net-Net SBC needs to be configured with an emergency LRT to route emergency calls to their destination.

As stated in the previous section, when searching an emergency LRT, any user defined parameter within a Location String may be used as the key to look up next-hop routing information.

LRT files support `<user type = "string">` which enables the Net-Net SBC to perform searches on free form attributes that may appear in the returned location-string. The `<user type = "string">` value for an entry in the emergency LRT should be set to a part or whole value returned in the CLF's location string. For example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<localRoutes>
  <route>
    <user type="string">1234</user>
    <next type="regex">!^.*$!sip:911@192.168.200.140:5060!</next>
  </route>
  <route>
    <user type="string">loc=xxx;noc=yyyy;line-code=zzzz</user>
    <next type="regex">!^.*$!sip:911@192.168.1.139:5060!</next>
  </route>
</localRoutes>
```

Note: Given that the Location String is not a well-defined string, care should be taken when defining and configuring the LRT tables.

Note: LRTs must be individually uploaded to both the active and standby systems in an HA node; LRTs are not automatically replicated across nodes.

CLF Response Failure

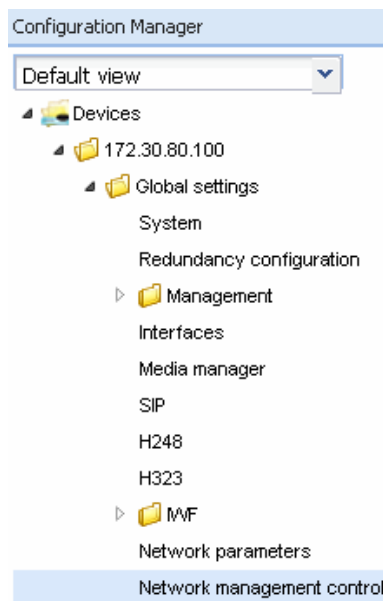
If there is no location string in a CLF's response or the CLF rejects the call, the Net-Net SBC uses the **default location string** parameter from the ingress SIP interface to populate the PANI header. The emergency call proceeds normally using this location string's information for emergency LRT lookups.

Configuring an NMC for E-CSCF Use

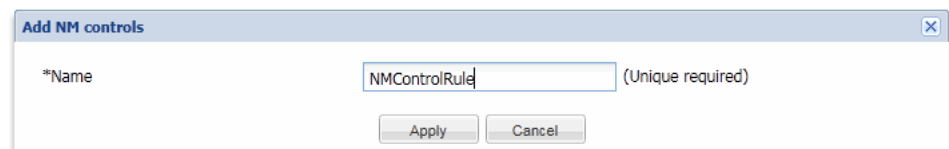
This procedure assumes that the Net-Net SBC is configured to communicate with a CLF. In addition, this procedure assumes the Net-Net SBC is configured and loaded with an appropriate LRT for E-CSCF use.

To configure an NMC for E-CSCF use:

1. Expand the Configuration Manager slider > Global settings.



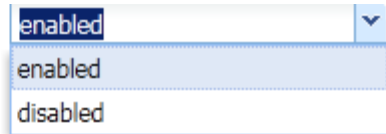
2. Expand the Global settings folder and click Network management control.
3. Click **Add**. The Add NM controls dialog box appears.
4. **Name**—Enter the name of this network management control rule; this value uniquely identifies the control rule. There is no default for this parameter.



5. Click **Apply**.

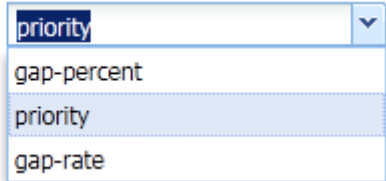
6. **State**—Select **enabled** from the combination box to enable this network management control rule. The valid values are **enabled** or **disabled**. The default value is **enabled**.

State

A dropdown menu with a blue border and a downward arrow on the right. The selected option is 'enabled' in a blue box. The menu is open, showing 'enabled' and 'disabled' as options.

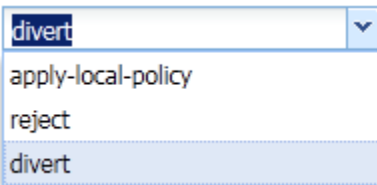
7. **Type**—Click **priority** in the combination box so that the Net-Net SBC will flag incoming calls with a matching **Destination ID** as a priority calls.

Type

A dropdown menu with a blue border and a downward arrow on the right. The selected option is 'priority' in a blue box. The menu is open, showing 'gap-percent', 'priority', and 'gap-rate' as options.

8. **Treatment**—Click **divert** in the combination box.

Treatment

A dropdown menu with a blue border and a downward arrow on the right. The selected option is 'divert' in a blue box. The menu is open, showing 'apply-local-policy', 'reject', and 'divert' as options.

9. **Next hop**—Leave this parameter blank so that the call's processing will go directly to the local policy.

Next hop

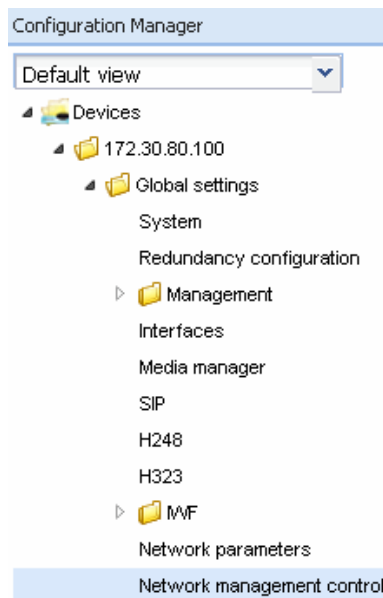
A simple rectangular text box with a thin border.

10. Click **Apply**.
11. Click **Apply**.

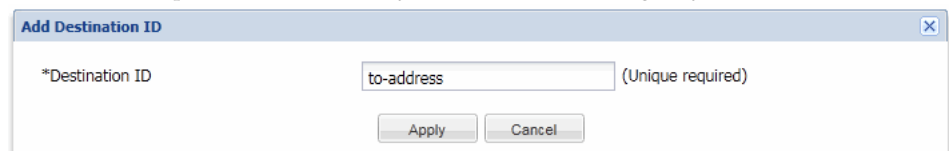
Configuring the Destination ID

To configure the destination ID:

1. Expand the Configuration Manager slider > Global settings.



2. Expand the Global settings folder and click Network management control.
3. Click a control rule in the NM controls table and click **Edit**.
4. Click **Add** next to **Destination ID**.
5. The Add DestinationId dialog box appears.
6. **Destination ID**—Enter the service URN that endpoints in your network include in their request URIs to identify themselves as emergency calls.

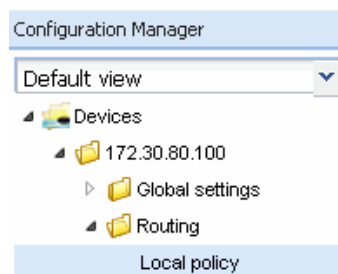


7. Click **Apply**. The destination ID is added to the Destination ID table.
8. Click **Apply**.

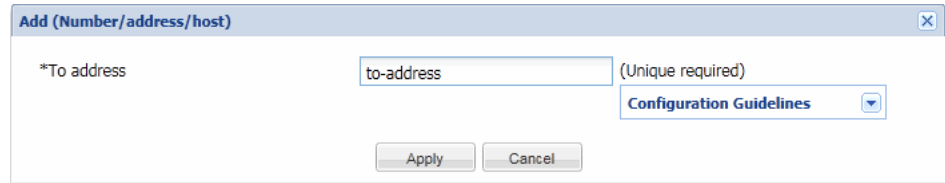
Configuring a Local Policy for E-CSCF Use

To configure a local policy for E-CSCF use:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Scroll to Request URI and click **Add**. The Add To address dialog box appears.
4. **Address**—Enter the lookup key for matching emergency calls. You can now use a service URN as lookup key criteria.



The dialog box titled "Add (Number/address/host)" contains a text input field labeled "*To address" with the placeholder text "to-address". To the right of the input field is a label "(Unique required)" and a button labeled "Configuration Guidelines" with a downward arrow. At the bottom of the dialog are two buttons: "Apply" and "Cancel".

5. Click Apply. The To address is added to the (number/address/host) list.
6. Click **Apply**.

Configuring Policy Attributes for E-CSCF Use

To configure policy attributes for E-CSCF you must select the ACLI view at the top of the configuration slider.

1. Expand the session-router folder and click local-policy.
2. Select the local policy you want to configure and click **Edit**.
3. Scroll to policy-attribute.
4. Click the policy attribute you want to configure and click **Edit**.
5. **eloc-str-lkup**—Click enabled in the drop-down list for the Net-Net SBC to parse the emergency location string, as received in a CLF Line Identifier AVP, for emergency LRT lookup.
6. **eloc-str-match**—Enter the attribute name found in the location string whose value will be used as a lookup key in the LRT named in the next-hop parameter. Common values include "loc" or "noc".
7. Click **Apply**.

SIP Surrogate Registration for Non-Register Requests

The Net-Net SBC's IMS functionality helps customers who use SIP IP PBX or SIP gateways that can only peer with carriers connected to IMS via a Proxy Call Session Control Function (P-CSCF). As part of this function, the Net-Net SBC provides for generating a Proxy-Authorization or Authorization header for REGISTER requests that are challenged. This feature extends the Net-Net SBC's capabilities by also allowing you to configure the generation of Proxy-Authorization and Authorization headers for non-REGISTER requests.

When you configured it to do so, the Net-Net SBC caches Proxy-Authorization or Authorization headers from the most recent (last-sent) messages in the following exchange: REGISTER--407 Proxy Authentication Required--REGISTER--200. Then the system uses these values in the subsequent requests. The following methods are supported:

- INVITE
- ACK
- BYE
- CANCEL

- UPDATE
- INFO
- PRACK
- OPTIONS

The Net-Net SBC updates the following parameters when it generates the header:

- nonce-count—Incremented for every new request the Net-Net SBC receives
- response—Contains the digest-request, newly generated using the cnonce, nonce, and other fields as input

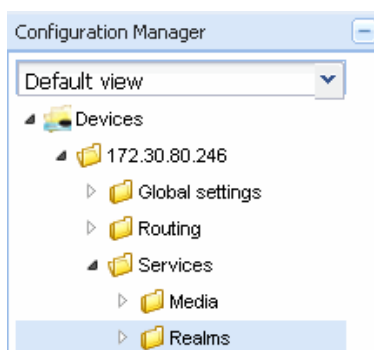
In addition, the system supports the noncetxt parameter in the Authentication-Info header. And for surrogate registration, it recognizes the Authentication-Info header in 200 OK responses received from the UAS and updates its cached nonce value accordingly; in this case, the system resets the nonce count to 1 for the subsequent request.

Configuring SIP Surrogate Registration

You configure this feature by setting two options in Surrogate agent in the Realms configuration: **auth-methods** and **auth-info**. To configure the methods you want used, you enter the option name followed by the methods you want supported.

To configure SIP surrogate registration:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm for which you want to edit a surrogate agent and click **Edit**.
4. Click the Surrogate registration tab at the top of the content area.
5. Click the surrogate agent you want to configure in the Surrogate agent table and click **Edit**.
6. Click **Options**. The Options dialog box appears.
7. Click **auth-methods** (or **auth-info**) in the Available options list to select it.
8. Click  to move the option to the Selected options list.

9. Repeat steps 6 and 7 if you want to select the second option (**auth-methods** or **auth-info**). The option name appears in the Selected options list along with a description in the lower part of the window.

The 'Options selector' dialog box is shown. It has two main panes: 'Available options' and 'Selected options'. Both panes contain a list with 'auth-info' and 'auth-methods'. Between the panes are '>>' and '<<' buttons. Below the panes, there is a section for the selected option 'auth-methods'. It includes a text box with the option name and a larger text area for the description: 'Generates a proxy authorization/authorization header for non-register requests made by surrogated users.' At the bottom, there is an 'Edit' button, a 'User defined option' text box, an 'Add' button, and 'OK' and 'Cancel' buttons.

10. **auth-methods**—Edit the textbox to modify the methods you want supported: **"invite, cancel ack, bye"**, separated by commas, and closed quotation marks (").

The 'Option' section of the dialog shows a text box containing the text: `auth-methods="invite,bye"`.

11. **auth-info**—Edit the textbox to append a label for the information.

The 'Option' section of the dialog shows a text box containing the text: `auth-info=refresh`.

12. Click **OK**.
13. Click **Apply**.

Introduction

The Net-Net SBC supports IP-to-IP transcoding for SIP sessions. It can connect two compressed voice streams that use different coding algorithms with one another.

This ability allows providers to:

- Handle the complexity of network connections and the range of media codecs that is endemic to VoIP with greater flexibility
- Optimize bandwidth availability by enforcing the use of high compression codecs
- Normalize traffic in the core network to a single codec
- Enact interconnection agreements between peer VoIP networks to use specified codecs

By providing transcoding capabilities at the network edge rather than employing core network resources for the same functions, the Net-Net SBC provides cost savings. It also provides a greater degree of flexibility and control over the codec(s) used in providers' networks and the network with which they interconnect.

In addition, placing the transcoding function in the Net-Net SBC and at the network edge means that transcoding can be performed on the ingress and egress of the network. The Net-Net SBC transcodes media flows between networks that use incompatible codecs. Transcoding eliminates back-hauling traffic to a centralized location, alleviating the need for multimedia resource function processors (MRFPs) and media gateways (MGWs) to support large numbers of codecs. Decreasing the need for codec support maximizes channel density usage for the MRFPs and MGWs so that they can reserve them for their own specialized functions.

Hardware Support: the Net-Net SBC TCU

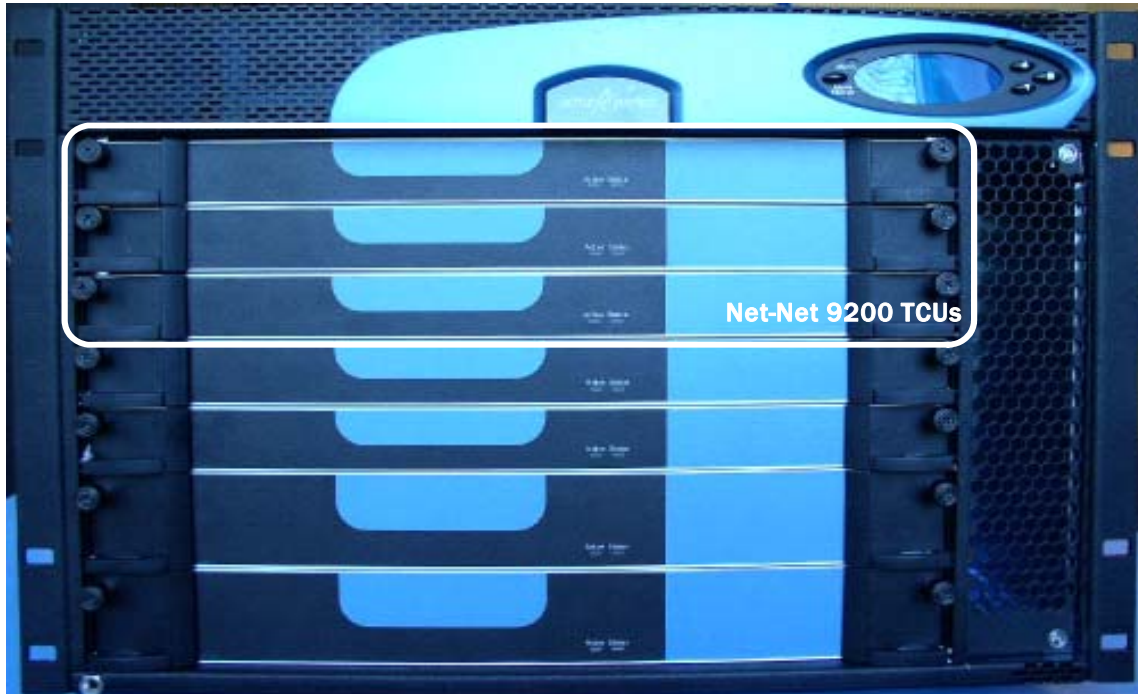
To use transcoding on your Net-Net SBC, you must have transcoding unit cards (TCUs) installed in your system. You can populate your Net-Net SBC with as few as one card or as many as three. It is possible to deploy one TCU alone, or to deploy TCU cards in either of the following redundancy schemes:

- 1+1—Active-standby mode for high availability
- 2+1—Requires three TCUs, where one card is a hot-standby for the two active cards; if one of the active cards fails, the hot-standby takes over, making failures transparent for call and media processing

The TCUs are installed into the front of the Net-Net SBC chassis. For more information about the TCU hardware and servicing the card, refer to the *Net-Net 9200 Hardware Guide*.

You do not need to obtain a license for this feature; transcoding is enabled automatically with your purchase of a TCU.

The following example shows you the location of the TCUs in the Net-Net SBC chassis.



Channel Density

Channel density depends on the following:

- Codecs used for transcoding
- Number of transcoding modules (TCMs) installed on the TCU; you can have between one and four TCMs per TCU
- Number of TCUs installed on the Net-Net SBC
- All system TCUs configured with the same number of TCMs

The following table shows the minimum channel density (bi-directional IP-IP media session; one TCU equipped with one TCM) per codec pairs.

	G.711	G.729A/B	G.729E	iLBC
G.711	2000			
G.729A/B	1050	750		
G.729E	530	380	380	
iLBC	530	380	380	380

The following table shows the maximum channel density per codec pairs, where there are two TCUs installed with four TCMs each.

	G.711	G.729A/B	G.729E	iLBC
G.711	8000			
G.729A/B	4200	3000		
G.729E	2120	1520	1520	
iLBC	2120	1520	1520	1520

How It Works

Because the Net-Net SBC sits between two networks, it can perform transcoding—allowing the entities with incompatible codecs to communicate.

The Net-Net SBC's transcoding capability is based on a SIP offer-answer model for negotiating SDP. An SDP offer can be in a SIP message such as an INVITE. The offer contains information about the codecs it would like to use. The responding network or device answers the SDP offer (answers appear in messages such as the SIP 200 OK) with its own set of supported codecs. If there are reINVITEs, then they are treated as new negotiations.

There are instances when the access and core network devices do not support the same codecs. The Net-Net SBC compares the codecs in the offer and answer, and if those codecs do not match, it enables its transcoding capability (if available). The Net-Net SBC generates a new offer media specification and returns it to the offering endpoint. If the Net-Net SBC does find matching codecs, then no transcoding is needed.

The Net-Net SBC changes the offer-answer in the model by supporting codec manipulations that add and delete codecs. The offer-answer model is broken into two-step negotiations. In transcoding policies, you can also specify the arrangement of codecs so that they are applied in a preferred order. Ingress and egress transcoding policies are determined by context:

- **Ingress**—An ingress transcoding policy is one specified for the offering side of a media negotiation. You can set ingress transcoding policies to act as a filter for certain codecs by configuring what the policy allows and what it does not.
- **Egress**—An egress transcoding policy is one specified for the answering side of a media negotiation; it is geared for the receiver.

These policies are configured and then applied to a realm, where you can set the name of the transcoding policy you want used for that realm. You can also transcode within the same realm.

About Media Formats

An incoming list of media formats (in the SDP contained in SIP messages) for each media stream contains:

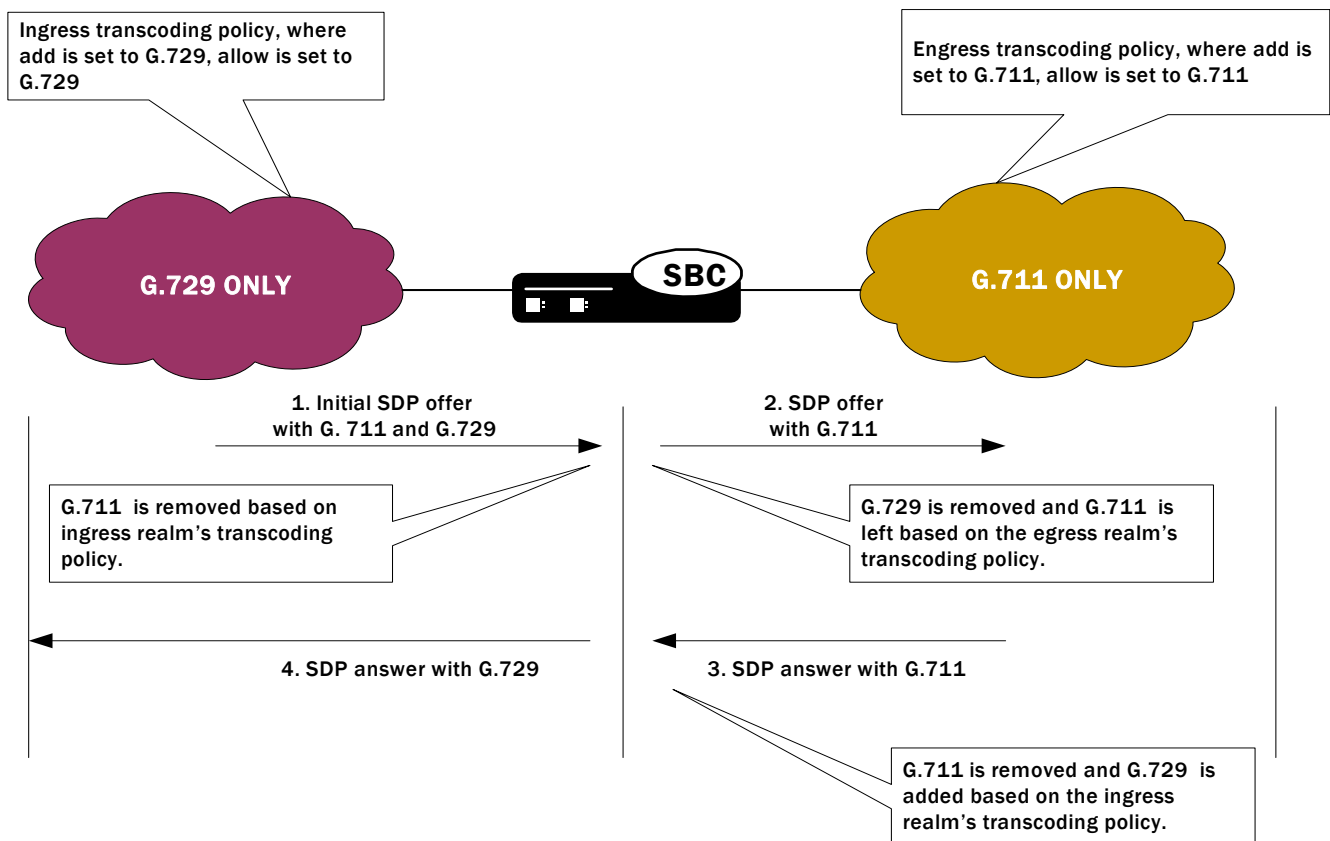
- A list of media formats, ordered by preference, that the offerer is capable of receiving; the media formats are codecs and any parameters associated with the codecs
- The RTP payload type numbers used to identify those media formats

If multiple formats are listed, it means that the offerer can receive any of those formats during the session.

The list of media formats to use for a session is contained in one of the sub-fields of the SDP media line (m=). For a given media format, there can be additional associated attribute lines (a=). The a= lines can provide dynamic binding of the media format to an RTP payload type, or additional attributes to further specify the codec. In order to manipulate the offer-answer, the Net-Net SBC changes m= lines and any other associated attributes as necessary.

Transcoding Example

In this example, the access network only allows codec G.729 because of bandwidth constraints, and thus must be normalized for communication with a core network that uses G.711. The callouts describe the transcoding policies that the Net-Net SBC is applying for the access and core realms, and the actions the systems are taking in their application.



Supported Codecs

The table in this section lists the codecs (with their bit rates and standard frame sizes) supported for Net-Net SBC transcoding.

Standard	Bit Rate	Native Frame Size
G.711	64 kbps	10 msec
G.729/A/B	8 kbps	10 msec
G.729e	8, 11.8 kbps	10 msec
iLBC	13.33 kbps 15.20 kbps	30 msec 20 msec

In addition, the following codecs are supported:

- DTMF RFC 2833
- T.38
- G.711 fallback (FAX)

Other possible values for codecs are wildcarding and none. You can wildcard codecs by entering an asterisk (*), or you can leave the setting at none.

The following table shows the allowable combinations that you can enter for transcoding policy parameters.

Allow Parameter Value	Add Parameter Value	Result
*	None	Any codec is allowed
*	Codec	Any codec is allowed; codecs in the Add list are added to the offer if they are not already present on the egress side
Codec	Codec	All codecs are removed except those specified in the Allow list; codecs in the Add list are added to the offer if they are not already present
Codec	None	All codecs are removed except those specified in the Allow list
None	Codec	All codecs are removed; codecs in the Add list are added to the offer if they are not already present

Optimized Transcoding Codec Pairs

In the Net-Net SBC, transcoding from any codec to another is possible; all transcoding is optimized with minimal processing and reduced latency.

EVRC Family of Codecs

The Net-Net 9000 supports 2 codecs in the EVRC family. Each codec has has three variants.

- EVRC-A is also commonly referred to as EVRC. The following EVRC-A packet formats are supported:

Header-freepacket format = EVRC0

Bundled/interleaved packet format = EVRC

Compact Budfled packet format = EVRC1

- The following EVRC-B packet formats are supported:

Header-freepacket format = EVRCB0

Bundled/interleaved packet format = EVRCB

Compact Budfled packet format = EVRCB1

EVRC-A Codec for Transcoding

EVRC Specifications

The Net-Net 9000 supports the Enhanced Variable Rate codec (EVRC) for transcoding. Three subtypes of the EVRC codec are supported as media-profiles:

1. EVRC0—header-free EVRC format. This codec is transcodable.
2. EVRC—non-header-free EVRC format. This codec is transcodable.
3. EVRC1—fixed rate EVRC format. This codec is transcodable.

To transcode to and from EVRC, the EVRC license must be installed.

Note: The Net-Net 9000 handles optional parameters according to RFC 4788 unless otherwise specified.

EVRC0 Supported Options

Required Parameters: none

Optional Parameters:

- **silencesupp**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.
- **dtxmax**—If this parameter is not present in a DTX session, the default value is 32.
- **dtxmin**—If this parameter is not present in a DTX session, the default value is 12.
- **hangover**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.

The payload type is dynamic for this codec. EVRC0 specifies the header-free format of the EVRC codec. Only a single frame (20 ms) is allowed in header-free mode.

EVRC Supported Options

Required Parameters: none

Optional parameters:

- **ptime**
- **maxptime**
- **maxinterleave**—If not signaled, the maxinterleave length is set to 5.
- **silencesupp**—If this parameter is not present, the default value 1 MUST be assumed.
- **dtxmax**—See dtxmax in EVRC0 section.
- **dtxmin**—See dtxmin in EVRC0 section.
- **hangover**—See hangover in EVRC0 section.

The payload type is dynamic for this codec. The default ptime is 20 ms. The ptimes 20, 40, and 60 ms are supported for transcoding.

EVRC1 Supported Options

Required parameters: none

Optional parameters:

- **ptime**—See RFC 4566
- **maxptime**
- **fixedrate**—Valid values are 0.5 or 1. If this parameter is not present, 0.5 is assumed.
- **silencesupp**—If this parameter is not present, 1 MUST be assumed.
- **dtxmax**—See dtxmax in EVRC0 section.
- **dtxmin**—See dtxmin in EVRC0 section.
- **hangover**—See hangover in EVRC0 section.

The payload type is dynamic for this codec. Only the 20 ms ptime is supported for transcoding. The media rate will be fixed to either full or half rate depending on the fixedrate parameter (half rate is default).

Default settings for EVRC encoding:

- CDMA Rate change for Dim and Burst disabled
- CDMA DTX control enabled

EVRC Configuration

Refer to EVRC codecs as follows:

EVRC0
EVRC
EVRC1

Refer to EVRC codecs' parameters exactly as they appear in the above list.

EVRC-B Codec for Transcoding

The Net-Net 9000 supports the Enhanced Variable Rate codec, extension B (EVRCB) for transcoding. Three subtypes of the EVRCB codec are supported as media-profiles:

1. EVRCB0—This codec is transcodable.
2. EVRCB—This codec is transcodable.
3. EVRCB1—This codec is transcodable.

To transcode to and from EVRCB, the EVRCB license must be installed.

Note: The Net-Net 9000 handles optional parameters according to RFC 4788 unless otherwise specified.

EVRCB0 Supported Options

EVRCB0 is the header-free format of the EVRCB codec. Only a single frame (20 ms) is allowed in header-free mode.

Required Parameters: none

Optional Parameters:

- **silencesupp**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.
- **dtxmax**—If this parameter is not present in a DTX session, the default value is 32.
- **dtxmin**—If this parameter is not present in a DTX session, the default value is 12.
- **hangover**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.

The payload type is dynamic for this codec. EVRCB0 specifies the header-free format of the EVRCB codec. Only a single frame (20 ms) is allowed in header-free mode.

EVRCB Supported Options

Required Parameters: none

Optional parameters:

- **ptime**
- **maxptime**
- **maxinterleave**—If not signaled, the maxinterleave length is set to 5.
- **silencesupp**—If this parameter is not present, the default value 1 MUST be assumed.
- **dtxmax**
- **dtxmin**
- **hangover**

The payload type is dynamic for this codec. The default ptime is 20 ms. Only the 20 ms ptime is supported for transcoding.

EVRCB1 Supported Options

Required parameters: none

Optional parameters:

- **ptime**—See RFC 4566
- **maxptime**
- **fixedrate**—Valid values are 0.5 or 1. If this parameter is not present, 0.5 is assumed.
- **silencesupp**—If this parameter is not present, 1 MUST be assumed.
- **dtxmax**
- **dtxmin**
- **hangover**

The payload type is dynamic for this codec. Only the 20 ms ptime is supported for transcoding. The media rate will be fixed to either full or half rate depending on the fixedrate parameter (half rate is default).

Default/fixed settings for EVRCB encoding:

- CDMA Rate change for Dim and Burst disabled
- CDMA Timing alignment disabled
- CDMA DTX control enabled
- EVRC-X Delay reduction disabled

EVRCB Configuration

In the ACLI, refer to EVRCB codecs as follows:

EVRCB0
EVRCB

EVRCB1

In the ACLI, refer to EVRC codecs' parameters exactly as they appear in the above sections in the bulleted lists.

EVRC Family of Codecs

The Net-Net 9000 supports 2 codecs in the EVRC family. Each codec has three variants.

- EVRC-A is also commonly referred to as EVRC. The following EVRC-A packet formats are supported:

Header-freepacket format = EVRC0

Bundled/interleaved packet format = EVRC

Compact Budfled packet format = EVRC1

- The following EVRC-B packet formats are supported:

Header-freepacket format = EVRCB0

Bundled/interleaved packet format = EVRCB

Compact Budfled packet format = EVRCB1

EVRC-A Codec for Transcoding

EVRC Specifications

The Net-Net 9000 supports the Enhanced Variable Rate codec (EVRC) for transcoding. Three subtypes of the EVRC codec are supported as media-profiles:

1. EVRC0—header-free EVRC format. This codec is transcodable.
2. EVRC—non-header-free EVRC format. This codec is transcodable.
3. EVRC1—fixed rate EVRC format. This codec is transcodable.

To transcode to and from EVRC, the EVRC license must be installed.

Note: The Net-Net 9000 handles optional parameters according to RFC 4788 unless otherwise specified.

EVRC0 Supported Options

Required Parameters: none

Optional Parameters:

- **silencesupp**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.
- **dtxmax**—If this parameter is not present in a DTX session, the default value is 32.
- **dtxmin**—If this parameter is not present in a DTX session, the default value is 12.
- **hangover**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.

The payload type is dynamic for this codec. EVRC0 specifies the header-free format of the EVRC codec. Only a single frame (20 ms) is allowed in header-free mode.

EVRC Supported Options

Required Parameters: none

Optional parameters:

- `ptime`
- `maxptime`
- `maxinterleave`—If not signaled, the `maxinterleave` length is set to 5.
- `silencesupp`—If this parameter is not present, the default value 1 MUST be assumed.
- `dtxmax`—See `dtxmax` in EVRC0 section.
- `dtxmin`—See `dtxmin` in EVRC0 section.
- `hangover`—See `hangover` in EVRC0 section.

The payload type is dynamic for this codec. The default `ptime` is 20 ms. The `ptimes` 20, 40, and 60 ms are supported for transcoding.

EVRC1 Supported Options

Required parameters: none

Optional parameters:

- `ptime`—See RFC 4566
- `maxptime`
- `fixedrate`—Valid values are 0.5 or 1. If this parameter is not present, 0.5 is assumed.
- `silencesupp`—If this parameter is not present, 1 MUST be assumed.
- `dtxmax`—See `dtxmax` in EVRC0 section.
- `dtxmin`—See `dtxmin` in EVRC0 section.
- `hangover`—See `hangover` in EVRC0 section.

The payload type is dynamic for this codec. Only the 20 ms `ptime` is supported for transcoding. The media rate will be fixed to either full or half rate depending on the `fixedrate` parameter (half rate is default).

Default settings for EVRC encoding:

- CDMA Rate change for Dim and Burst disabled
- CDMA DTX control enabled

EVRC Configuration

Refer to EVRC codecs as follows:

EVRC0
EVRC
EVRC1

Refer to EVRC codecs' parameters exactly as they appear in the above list.

EVRC-B Codec for Transcoding

The Net-Net 9000 supports the Enhanced Variable Rate codec, extension B (EVRCB) for transcoding. Three subtypes of the EVRCB codec are supported as media-profiles:

1. EVRCB0—This codec is transcodable.
2. EVRCB—This codec is transcodable.
3. EVRCB1—This codec is transcodable.

To transcode to and from EVRCB, the EVRCB license must be installed.

Note: The Net-Net 9000 handles optional parameters according to RFC 4788 unless otherwise specified.

EVRBC0 Supported Options

EVRBC0 is the header-free format of the EVRCB codec. Only a single frame (20 ms) is allowed in header-free mode.

Required Parameters: none

Optional Parameters:

- **silencesupp**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.
- **dtxmax**—If this parameter is not present in a DTX session, the default value is 32.
- **dtxmin**—If this parameter is not present in a DTX session, the default value is 12.
- **hangover**—If this parameter is not present in a DTX session, the default value 1 MUST be assumed.

The payload type is dynamic for this codec. EVRBC0 specifies the header-free format of the EVRCB codec. Only a single frame (20 ms) is allowed in header-free mode.

EVRCB Supported Options

Required Parameters: none

Optional parameters:

- **ptime**
- **maxptime**
- **maxinterleave**—If not signaled, the maxinterleave length is set to 5.
- **silencesupp**—If this parameter is not present, the default value 1 MUST be assumed.
- **dtxmax**
- **dtxmin**
- **hangover**

The payload type is dynamic for this codec. The default ptime is 20 ms. Only the 20 ms ptime is supported for transcoding.

EVRCB1 Supported Options

Required parameters: none

Optional parameters:

- **ptime**—See RFC 4566
- **maxptime**
- **fixedrate**—Valid values are 0.5 or 1. If this parameter is not present, 0.5 is assumed.
- **silencesupp**—If this parameter is not present, 1 MUST be assumed.
- **dtxmax**
- **dtxmin**
- **hangover**

The payload type is dynamic for this codec. Only the 20 ms ptime is supported for transcoding. The media rate will be fixed to either full or half rate depending on the fixedrate parameter (half rate is default).

Default/fixed settings for EVRCB encoding:

- CDMA Rate change for Dim and Burst disabled
- CDMA Timing alignment disabled
- CDMA DTX control enabled
- EVRC-X Delay reduction disabled

EVRCB Configuration

Refer to EVRCB codecs as follows:

EVRCB0

EVRCB

EVRCB1

Configuring Transcoding

This section describes how to configure transcoding policies on your Net-Net EMS.

When you configure transcoding on your Net-Net SBC, you want to consider:

- How to set up your ingress policies
- How to set up your egress policies
- To what realms you want to apply the transcoding policies you set up

Configuration Timestamp

You might see the following information on some of the configuration windows:

Last modified by **NNC_admin_10.1.20.11**

Last modified date **2011-04-14 15:31:46**

This is a configuration timestamp that displays last modified information for configuration changes made using Net-Net EMS.

When you make configuration changes to elements, the Last modified by field displays the following information:

- Net-Net EMS identifier
- Name of the user that made the modification
- IP address of the host machine upon which Net-Net EMS is running
- Relevant time zone information to specify how to interpret the timestamp in the Last modified date field.

For example:

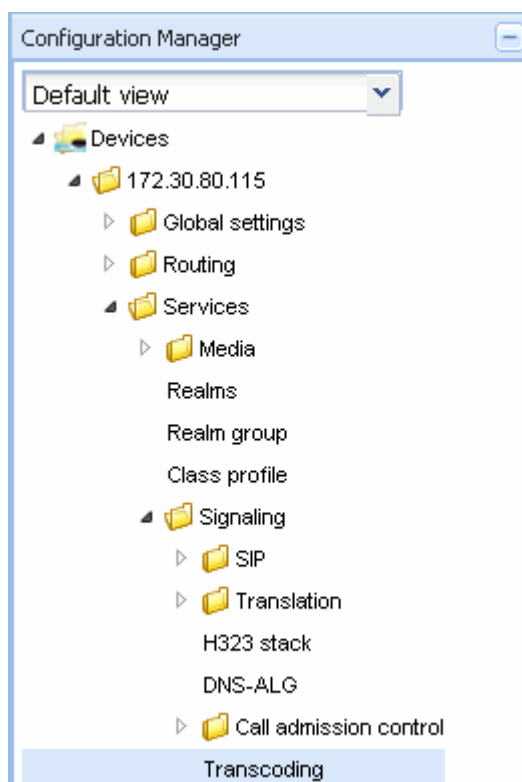
EMS_admin_172.30.80.2_UTC

The Last modified date is displayed in the string date/time format, YYYY-MM-DD HH:MM:SS. For example 2007-11-07 15:16:05. When you make configuration changes using the Net-Net EMS, the date timestamp is specified in UTC.

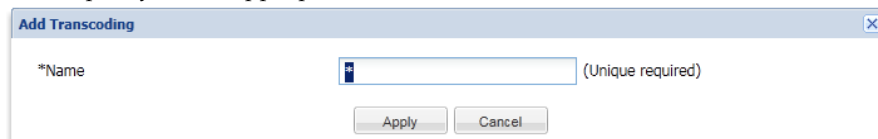
If a sub-element is modified the timestamp belonging to the element is updated to indicate the modification time.

Adding New Transcoding Policies

1. Expand the Configuration Manager slider > Services > Signaling



2. Click Transcoding. The Transcoding table appears in the content area.
3. Click **Add**. The Add Transcoding window appears.
4. **Name**—Enter the name you want to give this transcoding policy. The transcoding policy's name is important not only because it uniquely identifies the policy, but because it is the name you will enter into your realm configuration's Transcoding policy parameter. It is important to apply the correct policy to the appropriate realm..



5. Click **Apply**. The newly created transcoding policy appears in the table in the Transcoding window.
6. Select the new transcoding policy and click **Edit**.
7. **Force ptime**—Select enabled to enforce a ptime in the transcoding realm. The ptime inserted into the modified SDP will be equal to the frames per packet value multiplied by the native packetization interval of the preferred codec.
8. **Packetization time**—Enter a preferred ptime when the Force ptime parameter is enabled.

9. Click **Apply**.

Allowing, Adding, and Removing Codecs

The Net-Net SBC supports allowing, adding and removing codecs. Each of these operations works in the following way:

- **Allow**—The Allow codecs parameter takes a list of codecs that you want to pass through the Net-Net SBC; codecs not matching the items on this list are stripped.

You can wildcard this parameter to accept all supported codecs by simply entering an asterisk (*) as the parameter value.

An allowed codec can have a **forced** attribute, which is applicable only to an ingress transcoding policy. When you specify this attribute:

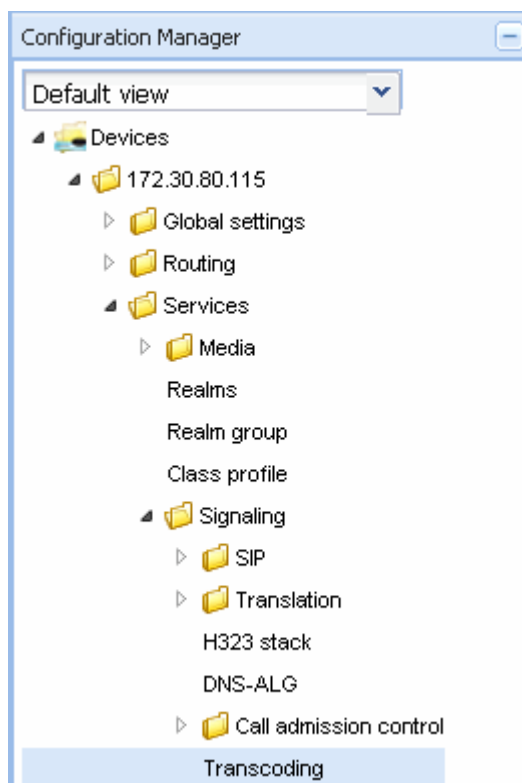
- The forced codec takes precedence over any non-forced matching codecs, even if a match is found in the answer; the forced codec (also matched) causes it to be used before the matched one(s)
 - When a codec specified as forced is present in the original offer, all non-forced codecs are removed
- **Add**—The Add codecs parameter sets the codecs that the Net-Net SBC adds to an offer if that codec is not already there. This parameter applies only to the egress policy, which is customized for the receiver. This is the conversion that the egress side requires for communication.

Allowing Codecs

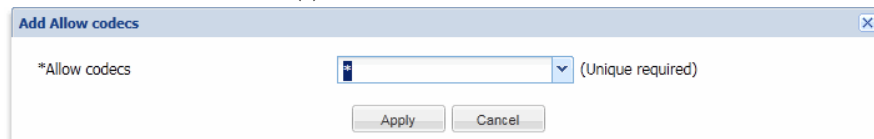
Codecs are added to the front of the codec list in the order specified. Added codecs are ignored if the top codec in the offer is not supported.

To allow codecs for transcoding policies:

1. Expand the Configuration Manager slider > Services > Signaling



2. Click Transcoding. The Transcoding table appears in the content area.
3. Select the transcoding policy you want to allow codecs for and click **Edit**. The Edit transcoding window appears.
4. In Allow codecs, click **Add**. The Add allow codecs window appears.
5. **allow codecs**—From the drop-down list, select a codec that is allowed to pass through the Net-Net SBC. If you leave this parameter set to **none** (default), then only codecs specified in the Add codecs parameter are included. To allow all codecs, enter an asterisk (*).

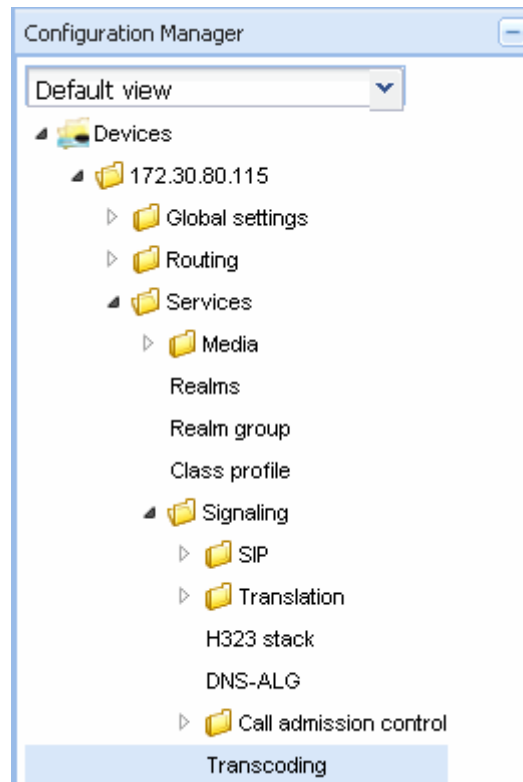


6. Click **Apply**.
Allowed codecs are added one at a time. To allow multiple codecs, repeat steps 3-5 for each.

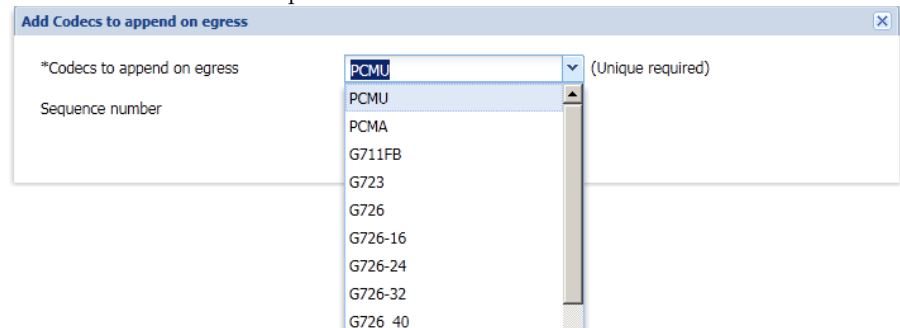
Adding Codecs

To add codecs to a transcoding policy:

1. Expand the Configuration Manager slider > Services > Signaling



2. Click Transcoding. The Transcoding table appears in the content area.
3. Select the transcoding policy you want to add codecs to and click **Edit**. The Edit transcoding window appears.
4. In Codecs to append on egress, click **Add**. The Add codecs on egress window appears.
5. **Codecs to append on egress**—Select the codecs that you want added to the SDP offer for the egress transcoding policy from the drop-down list. If you leave this parameter set to **none** (its default), then the Net-Net SBC will not add codecs to the offer. This parameter cannot be wildcarded.

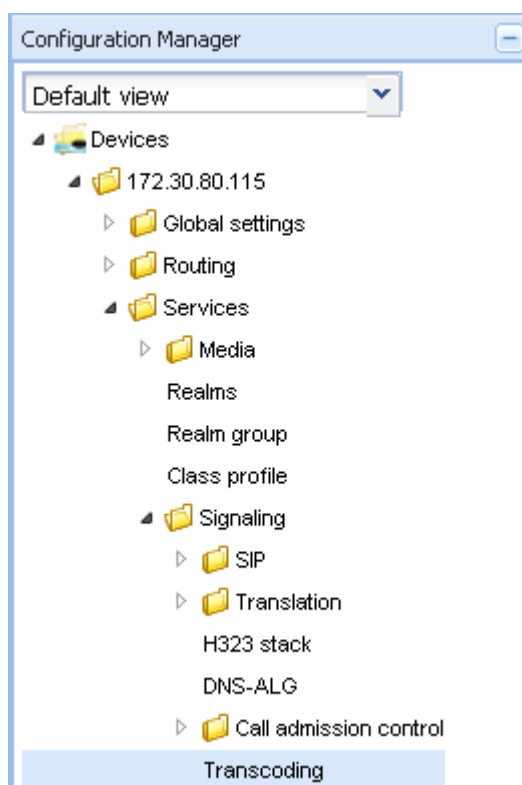


6. Click **Apply**.
Codecs must be added one at a time. To add multiple codecs, repeat steps 3-5 for each.

Removing Codecs

To remove codecs from a transcoding policy:

1. Expand the Configuration Manager slider > Services > Signaling



2. Click Transcoding. The Transcoding table appears in the content area.
3. Select the transcoding policy you want to remove codecs from and click **Edit**. The Edit transcoding window appears.
4. In Codecs to append on egress, select the codec you want to delete and click **Delete**. A confirmation dialog box appears asking if you really want to delete the codec.
5. **Confirmation**—Click **Yes** to finish deleting the codec.

Specifying an Order for Codecs

In your transcoding policy, you can specify the order of preference for codecs in the offer.

Codecs can be ordered in different ways. Your order listing cannot contain a single codec, nor can it contain a single wildcard value (*). A single wildcard value tells the Net-Net SBC to use the original order. However, you can use the wildcard in the list that you create. If there are any remaining codecs that do not match the order you enter, they are appended to the end of the list.

The following example uses letters as placeholders to show how you can manipulate the order.

Given codecs A B C D E F G:

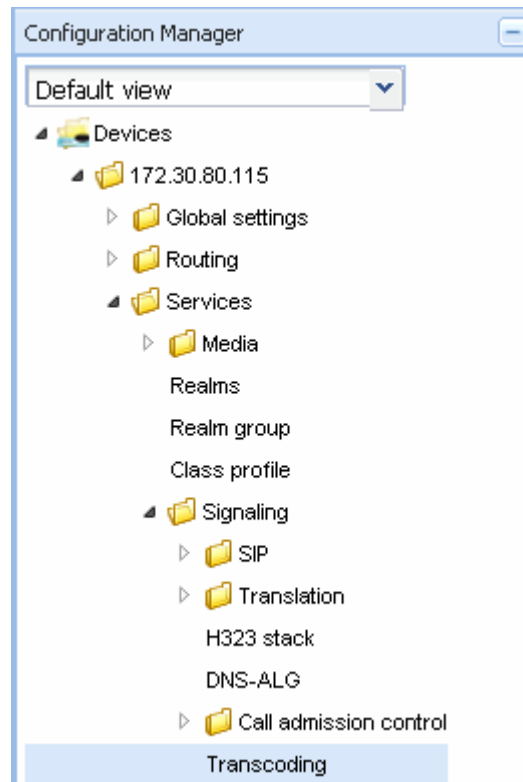
- You can specify the order of each codec in the list. For example, you might change the order listed above to: F E D G B A and C.

- You can specify the order of part of the list, and leave the rest to be appended in the order they appear. You might only specify A D E and G. The order would then be A D E G C F. C and F are the codecs not listed, and are added to the end of the list in their original sequence: C comes first, then F.
- You can place a wildcard value by entering an asterisk (*) at the beginning, in the middle, or at the end of the ordering.
 - If you want codecs D and E to appear last in the order, you would enter *DE. The resulting order would then be A B C F G D E.
 - If you want codecs D and E to appear in the first and last positions in the order, you would enter D*E. The resulting order would then be D A B C F G E.
 - If you want codecs D and E to appear first in the order, you would enter DE*. The resulting order would then be D E A B C F G.

Configuring Codec Order

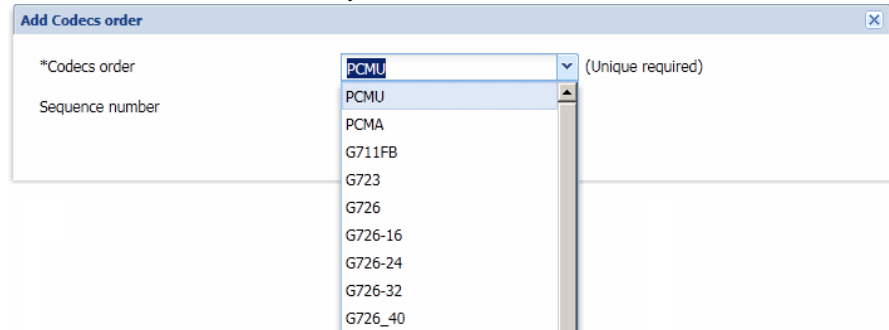
To configure an order of preference for codecs in the offer:

1. Expand the Configuration Manager slider > Services > Signaling

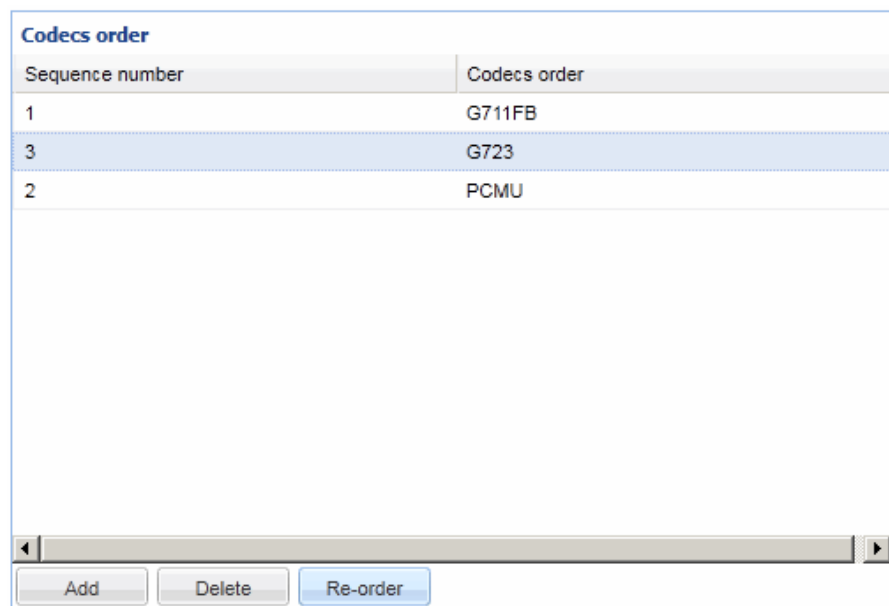


2. Click Transcoding. The Transcoding table appears in the content area.
3. Select the transcoding policy you want to specify codec order for and click **Edit**.
4. In Codecs order, click **Add**. The Add order codecs window appears.

5. **Order codecs**—Specify the codec order by selecting codecs from the combination box. You can only select one codec at a time.



6. Click **Apply**. The codec appears in the Codecs order table. You can add multiple codecs by adding them one at a time.
7. Change the order of the codecs by selecting each and dragging it to where you would like it.



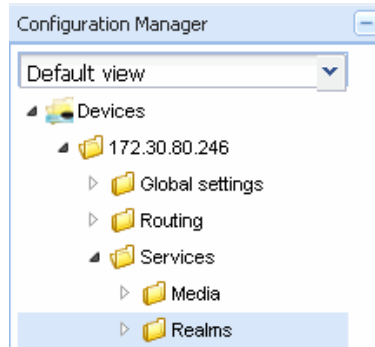
8. Click the Re-order button to submit the new order of the codecs.

Applying a Transcoding Policy to a Realm

Once you have configured the transcoding policies that you need, you apply them to realms. You need to know the name you gave to transcoding policy so that you can enter it into a realm configuration.

To apply a transcoding policy to a realm:

1. Expand the Configuration Manager slider > Services.



2. Click Realms. The Realm table appears in the content area.
3. Select the realm you want to apply a transcoding policy to and click **Edit**.
4. **Transcoding policy**—Select the transcoding policy you want to apply to this realm from the drop-down list. There is no default for this parameter. .

Transcoding policy

Transcoding in the Same Realm

If you want transcoding to take place within the same realm, you can enable that function in the Edit Realm window.

Note: Acme Packet recommends that you leave the **within the single realm** parameter unchecked if your realm configuration's **Within this realm serviced by this Net-Net system only** parameter is also left unchecked.

To transcode within a realm:

1. In the Realm window, select the realm whose transcoding policy you would like to edit. Click **Edit**.
2. **Within the single realm**—Click the checkbox to enable transcoding within the same realm.

Within the single realm

Transcoding in the Same Network

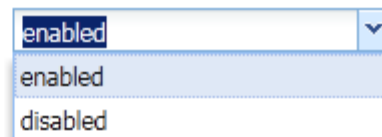
If you want transcoding to take place within the same network, you can enable that function in the realm configuration.

To transcode within a network:

1. In the Realm window, select the realm whose transcoding policy you would like to edit and click **Edit**.

2. **Within the same network**—Click the checkbox to enable transcoding within the same network.

Within the same network



3. Click **Apply** to continue.

AMR-NB and EVRC0 Transcodable Codecs

AMR-NB

The Net-Net SBC supports Adaptive Multi-Rate Narrow Band (AMR-NB) codec. All configurations of this codec, as indicated by SDP, are transcodable except when the following SDP parameters are enabled:

- robust sorting
- interleaving

When AMR is configured in a transcoding policy's **Codecs to append on egress** parameter, it is forwarded from the Net-Net SBC with the following default settings:

- 12.2 kbps
- RTP/IF1 format
- No redundant packets
- Bandwidth efficient payload
- No CRC frame
- 20ms default ptime

Note: This codec requires a license.

EVRC0

The Net-Net SBC supports the header-free version of Enhanced Variable Rate codec (EVRC) for transcoding.

The Net-Net SBC is configured with two versions of the EVRC media-profile:

- EVRC—non-header-free EVRC format. This codec is NON-transcodable.
- EVRC0—header-free EVRC format. This codec is transcodable.

When EVRC0 is configured in a transcoding policy's **Codecs to append on egress** parameter, it is forwarded from the Net-Net SBC with the following default settings:

- Header-Free packet format
- CDMA Rate change for Dim and Burst disabled
- CDMA Timing alignment disabled
- CDMA DTX control disabled
- EVRC-X Delay reduction disabled
- 20ms default ptime

Note: This codec requires a license.

Removing, Allowing, and Adding Codecs

The Net-Net SBC removes and allows codecs using the **allow** parameter.

- **Allow codecs**—Takes a list of codecs that you want to pass through the Net-Net SBC and can explicitly allow them to remain in the SDP for the next step; codecs not matching the items on this list are removed. This parameter is required.
- **Codecs to append on egress**—Sets the codecs that the Net-Net SBC adds to an offer if that codec is not already there. This parameter applies only to the egress policy.

Transcoding Capacity

Transcoding capacity depends on the following:

- Codecs used for transcoding
- Number of transcoding modules (TCMs) installed on the TCU; you can have between one and four TCMs per TCU
- Number of TCUs installed on the Net-Net 9200

The following table shows minimum session capacity (bi-directional IP-IP media session; one TCU equipped with one TCM) per codec pair with the following assumptions:

- 20ms packetization period or higher
- 100% use of the code pair identified by the corresponding column/row header
- As you reach the channel density limit, the session establishment rate will impact the gradient at which this limit will be reached, at high call rates there may be some sessions rejected before the limit is achieved
- EVRC-B is NOT licensed

Transcoding Capacity per TCM:

	G.711	G.726	G.729A/B	G.723	ILBC	G.728	GSM-FR	AMR-NB	EVRC	EVRCB	T.38
G.711	1000										
G.726	1000	1000									
G.729A/B	1000	890	760								
G.723	910	790	690	630							
ILBC	610	550	500	470	380						
G.728	500	460	420	400	330	290					
GSM-FR	870	760	660	610	460	390	590				
AMR-NB	800	710	620	580	440	380	560	530			
EVRC	740	650	580	540	420	360	520	500	470		
EVRCB	Not Licensed										
T.38	50										

Each TCM has a hard limit of 50 T.38 transcoding sessions. This is in addition being constrained by available processing resources that may be used by other transcoding tasks. Therefore a fully populated TCU can support 200 transcoded FAX calls, provided enough DSP resources are available.

Transcoding Capacity per TCM with EVRC-B licensed:

If EVRC-B is licensed on the system, there is an transcoding capacity impact for certain pairs (as highlighted).

	G.711	G.726	G.729A/B	G.723	iLBC	G.728	GSM-FR	AMR-NB	EVRC	EVRCB	T.38
G.711	1000										
G.726	1000	1000									
G.729A/B	1000	880	760								
G.723	910	790	680	630							
iLBC	610	550	500	470	370						
G.728	490	450	420	400	330	290					
GSM-FR	870	750	660	610	460	390	590				
AMR-NB	800	700	620	570	440	370	560	530			
EVRC	730	650	580	540	420	360	520	500	470		
EVRCB	530	480	440	420	340	300	410	390	380	320	
T.38	50										

For estimating specific transcoding hardware requirements to support your network deployment, please contact your System Engineers and sales team.

Licensing

The following licenses are required to use this feature:

- Transcode Codec AMR
- Transcode Codec EVRC

Media Type Subnames

You can define multiple versions of a media profile for a single codec by using subnames. You can then reference the new media profile by a combination of the media profile name and media profile subname.

Some media types are not unique per just their value in an SDP **m=** line, they must be uniquely identified by looking at additional SDP parameters. For example, you can define a media profile for G729, when only the parameter and value **annexb=yes** is present in the SDP. By creating a media profile + subname that defines both a media type and parameter, you can perform various operations on G729 only when **annexb=yes** is encountered.

Some applications of media type subnames are:

- maintaining different versions of the same codec with different bandwidth ceilings
- maintaining different versions of the same codec with different ptimes
- grouping codecs by using customer as a subname
- grouping codecs by using realm as a subname

SDP Parameter Matching

This feature matches parameters in the **a=fmtp**, codec-specific SDP **a=** line. It does not try to match a global **m=** line attribute like **a=p-time**.

Using Subnames with Transcoding Policies

If no subname has been created for a media profile, you can continue using the media profile name without any subname specifier.

Subname Syntax and Wildcarding

You can wildcard one or both portions (name and subname) of a media type and subname pair:

- When you wildcard the **name** portion of the value, you can provide a specific subname that the Net-Net SBC uses to find matching media profiles.
- When you wildcard the subname portion of the value, you can provide a specific name that the Net-Net SBC uses to find matching media profiles.

The following table defines and explains subname wildcarding and syntax:

Syntax	Example Value	Description
<name>	PCMU	Matches any and all media profiles with the name value configured as PCMU. This entry has the same meaning as a value with this syntax: <name>::*.
<name>::	PCMU::	Matches a media profile with the name with the name value configured as PCMU with an empty subname parameter.
<name>::*	PCMU::*	Matches any and all media profiles with the name value configured as PCMU with any and all subname configured.
<name>::<subname>	PCMU::64k	Matches a media profiles with the name with the name value configured as PCMU with the subname parameter set to 64k.
*	*	Matches anything, but does not have to be a defined media profile.
::	*::*	Matches any and all media profiles, but requires the presence of media profile configurations.
*::<subname>	*::64k	Matches all media profiles with this subname. You might have a group of media profiles with different names, but the same subname value.
*::	*::	Matches any media profiles with an empty subname parameter.
::	::	Invalid
::*	::*	Invalid

Wildcarding add on egress

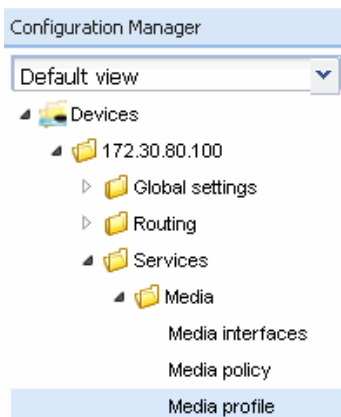
You cannot configure **add on egress** with a wildcarded subname in a transcoding policy. You can only add a specific instance of a media type.

Configuring a Media Type Subname

To use media type subnames with a transcoding policy, you must first configure a media profile and subname. Then you can configure a transcoding policy with a media type and subname pair for your application.

To configure a media type subname:

1. Expand the Configuration Manager slider > Services > Media.



2. Click Media profile. The Media profile table opens in the content area.
3. Click **Add**. The Add Media profile dialog box appears.
4. **Name**—Enter the encoding name used in the SDP rtpmap attribute.
5. **Subname**—Enter a subname to create multiple profiles with the same codec name.

 The screenshot shows a dialog box titled 'Add media-profile'. It has two input fields: '*name' with the value 'G728' and '*subname' with the value 'annexa-yes'. Both fields have '(Unique required)' text to their right. At the bottom are 'Apply' and 'Cancel' buttons.

6. Click **Apply**.

AMR-NB and EVRC0 Transcodable Codecs

AMR-NB

The Net-Net SBC supports Adaptive Multi-Rate Narrow Band (AMR-NB) codec. All configurations of this codec, as indicated by SDP, are transcodable except when the following SDP parameters are enabled:

- robust sorting
- interleaving

When AMR is configured in a transcoding policy's **Codecs to append on egress** parameter, it is forwarded from the Net-Net SBC with the following default settings:

- 12.2 kbps
- RTP/IF1 format
- No redundant packets
- Bandwidth efficient payload
- No CRC frame
- 20ms default ptime

Note: This codec requires a license.

EVRC0

The Net-Net SBC supports the header-free version of Enhanced Variable Rate codec (EVRC) for transcoding.

The Net-Net SBC is configured with two versions of the EVRC media-profile:

- EVRC—non-header-free EVRC format. This codec is NON-transcodable.
- EVRC0—header-free EVRC format. This codec is transcodable.

When EVRC0 is configured in a transcoding policy's **Codecs to append on egress** parameter, it is forwarded from the Net-Net SBC with the following default settings:

- Header-Free packet format
- CDMA Rate change for Dim and Burst disabled
- CDMA Timing alignment disabled
- CDMA DTX control disabled
- EVRC-X Delay reduction disabled
- 20ms default ptime

Note: This codec requires a license.

Removing, Allowing, and Adding Codecs

The Net-Net SBC removes and allows codecs using the **allow** parameter.

- **Allow codecs**—Takes a list of codecs that you want to pass through the Net-Net SBC and can explicitly allow them to remain in the SDP for the next step;

codecs not matching the items on this list are removed. This parameter is required.

- **Codecs to append on egress**—Sets the codecs that the Net-Net SBC adds to an offer if that codec is not already there. This parameter applies only to the egress policy.

Transcoding Capacity

The following matrix displays the updated transcoding capacity per TCM.

	G.711	G.726	G.729A/B	G.723	G.729E	iLBC	G.728	GSM FR	AMR	AMRWB	EVRC
G.711	1000										
G.726	1000	1000									
G.729A/B	1000	906	775								
G.723	932	806	701	640							
G.729E	622	564	510	477	380						
iLBC	622	564	510	477	380	380					
G.728	503	464	427	404	332	332	295				
GSM FR	889	774	676	619	465	465	396	600			
AMR	821	722	637	586	446	446	382	568	540		
AMRWB	511	471	433	409	335	335	297	400	386	300	
EVRC	750	666	593	549	424	424	365	533	508	369	480
T.38	50										

Licensing

The following licenses are required to use this feature:

- Transcode Codec AMR
- Transcode Codec EVRC

H.323 - G726 Transcoding

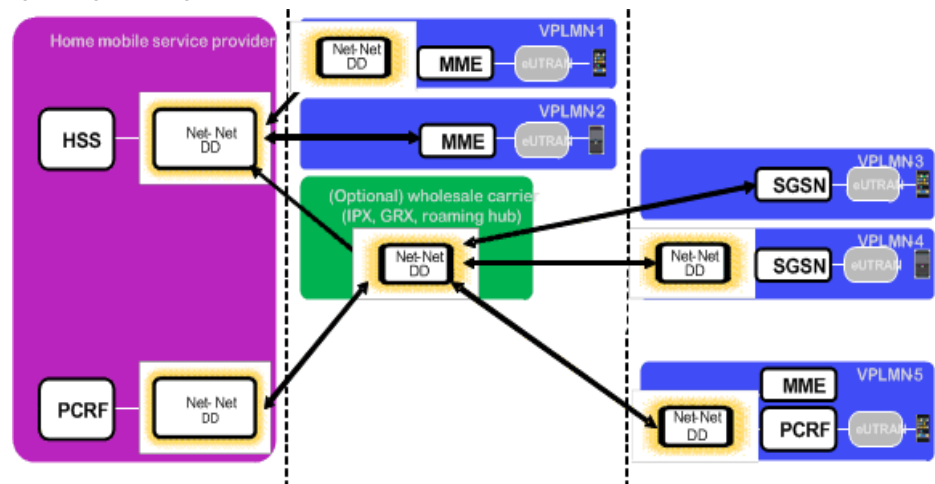
You can now configure your D-Series SBC to support H.323 - G726 transcoding.

Diameter Network Architecture

Network Architecture

The Net-Net Diameter Director participates at a border point between Public Land Mobile Networks (PLMNs, or Mobile Service Providers) which is created within the LTE data deployments when roaming.

When a roaming subscriber attempts to use the VPLMN (Visited PLMN/Visited MSP) Diameter signaling messages are exchanged between the Mobility Management Entity (MME) or SGSN (Serving GPRS Support Node) and Home Subscriber Server (HSS) in order to provide authentication and subscription authorization information about the roaming subscriber. When policy control dips are required to authorize data services, the visited PCRF (Policy and Charging Rules Function) contacts the home PCRF via Diameter signaling messages. The high level topology and relationship of these logical elements that are used to exchange signaling messages is shown below.



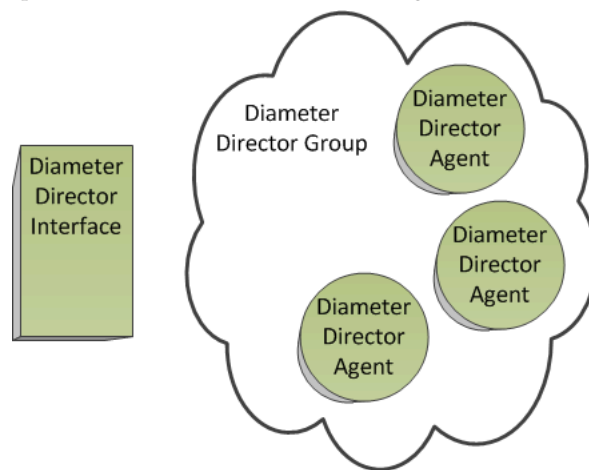
The Net-Net Diameter Director provides control for both the MME / SGSN to HSS and PCRF to PCRF signaling messages at this data border between Home PLMNs, Visited PLMNs, and the wholesale carriers that (optionally) connect the PLMNs together.

20 Diameter Director Configuration

Diameter Director Elements

As the Net-Net Diameter Director sits in a network, the two primary elements that must be configured on the system are the Diameter Director Interface and the Diameter Director Agent. In addition, a Diameter Director Group can be created as a virtual collection of Diameter Director Agents.

The Diameter Director Interface is the Diameter application interface that runs on the Net-Net Diameter Director, while the Diameter Director Agent is the representation of remote Diameter agent.



Diameter Director Interface

The Diameter Director Interface is the Diameter application interface that runs on the Net-Net Diameter Director. Since there can only be one Diameter Director Interface active in a realm, it is defined by that realm, which is configured in the *realm-id* parameter.

When a message is received on a Diameter Director Interface, the Net-Net Diameter Director then determines how and where to route it. Each Diameter Director Interface is configured with a root Diameter Director Policy that sets the starting point of the routing process. See the [Routing Diameter Messages \(1681\)](#) chapter.

Supported Applications

Each Diameter Director Interface may be configured with a set of Diameter Application IDs and Vendor IDs. These values are used and sent in a capability exchange with external Diameter agents to determine the range of Diameter applications that each network element supports (See: [Capabilities Exchange Messaging \(1677\)](#)).

The Application IDs configured at the Diameter Director Interface level are the least specific Application ID for the Diameter Director Interface to Diameter Director Agent connection. Thus if an Application ID is configured for either the Diameter Director Group or Diameter Director Agent, they take precedence over the Application IDs configured on the Diameter Director Interface.

In addition, a list of supported vendor IDs can be configured in the root Diameter Director Interface. These configured values are included in the CER/CEA negotiation as Supported-Vendor-ID AVPs.

Origin Host

The Origin-Host AVP (AVP Code 264) is present in all Diameter messages. This AVP identifies the endpoint that originated the Diameter message. When acting as a relay, the Net-Net Diameter Director will not modify this AVP.

The default Net-Net Diameter Director behavior sets the Origin-Host AVP as <Diameter director interface IP address>.<realm-id> in all outgoing messages. The value in the Origin-Host AVP is effectively ignored.

The Origin Host AVP contents can also be configured by setting the *origin host identifier* parameter in the *diameter director interface* configuration element. In addition to this the format of the value must be specified with the *origin host format* parameter. The origin host format parameter accepts the following enumerated values:

- None—The default PD behavior is retained which sets the Origin-Host AVP as <interface-ip-addr>.<realm-id> in all outgoing messages.
- Identifier—The Origin-Host AVP in all outgoing message will be set to the value of the *origin-host-identifier* parameter.
- Identifier-with-realm—The Origin-Host AVP in all outgoing messages will be set to <origin-host-identifier>.<realm-id>.

Diameter Director Ports

Each Diameter Director Interface requires at least one Diameter Director Port subelement. The Diameter Director Port defines the actual IP address/port and transport protocol where a Diameter agent may connect to on a Net-Net Diameter Director and where messages are sent and received. To define this, the *address*, *port* and *transport protocol* are configured. No two Diameter Director ports subelements may use the same IP address. In addition, admission control settings can be defined for each Diameter Director Port (see: "[Anonymous Diameter Agent Blocking](#)").

SCTP

The Net-Net Diameter Director supports SCTP as the transport protocol between a Diameter Director Interface and a Diameter agent. Set the *transport protocol* parameter to **sctp** within a *diameter director ports* subelement to use SCTP. The corresponding Diameter Director Agent must also have its *transport protocol* parameter set to **sctp** as well.

Licensing

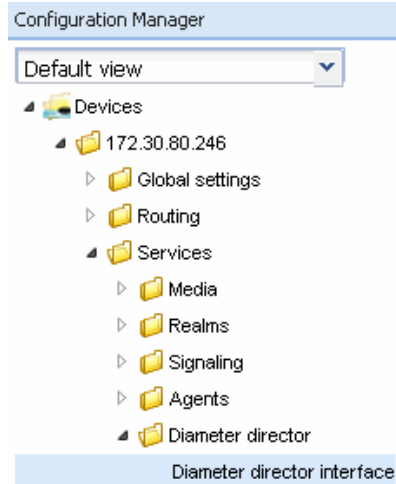
The Net-Net Diameter Director requires a "Diameter Director" license to unlock all relevant configuration elements, statistics, and let the Diameter Director task begin running. Please ensure that you have this license installed on your system.

Note: If the license was installed after the Net-Net Diameter Director runtime image was loaded, you will need to reboot the system. The Diameter Director license must be installed in the system prior to boot time.

Configuring Diameter Director Interfaces

To configure a Diameter Director interface:

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter Director interface. The Diameter Director interface table appears in the content area.
3. Click **Add**.
4. **Realm**—Enter the realm name where this Diameter Director Interface exists.
5. **Address**—Enter an IP address in the realm where this Diameter Director Interface can send and receive messages.
6. **Port**—You can override the default port (3868) by setting this to a value of your choosing.
7. **Transport protocol**—Set this parameter to either TCP or SCTP to indicate the transport protocol.

Add Diameter director interface ✕

*Realm

arlington98

▼ (Unique required)

Diameter director ports

*Address

172.30.80.73

(Enter a valid IP address)

*Port

3868

(Range:1025..65535.
Default:3868)

*Transport protocol

TCP

▼ (Default:TCP)

Apply

Cancel

8. Click **Apply**. The new Diameter Director interface now appears in the table.

Configuring Diameter Director Ports

To configure the Diameter Director ports:

1. Select the Diameter Director interface from the table and click **Edit**.
2. Scroll to the Diameter Director ports table and click **Add**.
3. **Address**—Enter an IP address in the realm where this Diameter Director Interface can send and receive messages.
4. **Port**—You can override the default port (3868) by setting this to a value of your choosing.
5. **Transport protocol**—Set this parameter to either TCP or SCTP to indicate the transport-protocol. This Diameter Director port subelement will now only use the indicated transport protocol.
6. Click **Apply**. The Diameter Director port now appears in the table.
7. Select the port from the table and click **Edit**.
8. **Anonymous mode**—See: Anonymous Diameter Agent Blocking.

*Address	100.20.30.40
*Port	3868
*Transport protocol	TCP
Anonymous mode	all agents-only all

9. Click **Apply**.
10. Click **Back**.

Configuring Supported Vendor IDs, Application IDs and their Vendor IDs

To configure Supported Vendor IDs and Application IDs and their Vendor IDs:

1. Scroll to the Supported Vendor IDs table and click **Add**.
2. **Supported vendor IDs**—Enter the Vendor ID values to include in the CER/CEA exchange.
3. Click **OK**.

Add

OK Cancel

4. Scroll to the Diameter Director application table and click **Add**.
5. **Application ID**—Select an Application ID that this Diameter Director Interface supports from the drop-down list.
6. Click **Apply**.
7. Select the Application ID from the table and click **Edit**.
8. **Application type**—Set this to **accounting** or **authentication** depending on the application type.

9. **Vendor ID**—Enter the Vendor ID for this application ID. (Enter 0 for relay).

*Application ID	16777229	▼
Application type	accounting	▼
Vendor ID	10415	

10. Click **Apply**.
11. Click **Back**.

Configuring Origin Host AVP Appearance

To configure how the Origin Host AVP appears in CER/CEA negotiation:

1. **Origin host ID**—Enter a unique identifier to use in the origin host AVP to override the Net-Net Diameter Director's default method of creating an Origin Host AVP.
2. **Origin host format**—Select **identifier** or **identifier-with-realm** from the drop-down list to indicate how to create the Origin Host AVP.
3. Click **Apply**.

Diameter Director Agent

The Diameter Director Agent is the representation of a remote Diameter agent. For all Diameter agents that may connect to the Net-Net Diameter Director, a Diameter Director Agent must be created. Diameter Director Agents must be in the same realm as a Diameter Director Interface to communicate. If there is no Diameter Director Interface that a Diameter Director Agent can point to, no connection will be made.

Like a Diameter Director Interface, a Diameter Director Agent is configured with supported Application IDs to assert in a CER/CEA negotiation. The difference is that the group of Application IDs configured on an Diameter Director Agent supersede those configured on a Diameter Director Interface and Diameter Director Group. To inherit the Application IDs configured on the corresponding Diameter Director Interface or Diameter Director Group, do not configure a *diameter director application* subelement on a Diameter Director Agent.

A Diameter Director Agent is configured with a hostname that is used as a reference within the Net-Net Diameter Director, and primarily in the Diameter Director Group configuration element or a *next-hop* in a Policy Attribute in a Diameter Director Policy.

The *ip address*, *port*, *realm id* and *transport protocol* are standard configurations for an external agent. To use an FQDN via DNS for reaching a Diameter Director Agent, you may configure the *hostname* parameter and leave the *ip-address* parameter blank (supporting DNS services must also be configured. The *transport protocol* must match the *transport protocol* located in the *diameter director ports* subelement on the Diameter Director Interface where it connects. Available transport protocols for Diameter messaging are TCP and SCTP.

While each possible Diameter Director Agent must be created as a *diameter director agent* configuration element, the *connection-mode* determines whether the Net-Net Diameter Director connects first to the Diameter Director Agent with the **outbound** value or waits for an incoming connection from the agent with the **inbound** value.

Any time a Diameter connection is lost between the Net-Net Diameter Director and a Diameter agent with a *connection-mode* of **outgoing**, the Net-Net Diameter Director begins its process of reconnecting with the agent. See [Anonymous Diameter Agent Blocking \(1702\)](#) for additional information on the security aspects of this feature.

Watchdog Messages

All Diameter Director Interfaces reply to Device Watchdog Requests (DWR) issued to them with a Device Watchdog Answers (DWA).

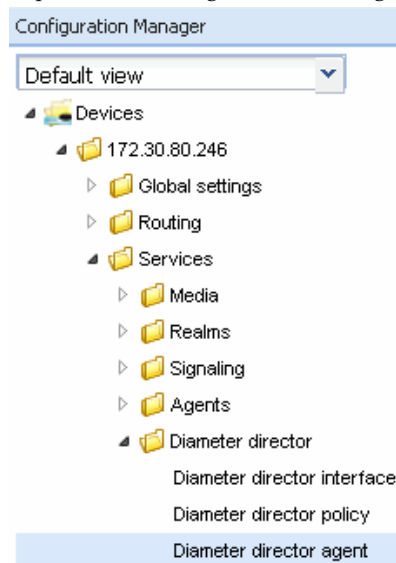
To configure the Net-Net Diameter Director to initiate DWRs toward a Diameter agent, the *watchdog-timer* parameter is configured in the *diameter director agent* configuration element. When set to a non-0 value, this parameter controls how often to send DWRs to a Diameter agent. The Net-Net Diameter Director's internal timer is reset every time it receives a Diameter message. If the watchdog timer expires due to inactivity on a connection, a DWR is sent to the remote agent. If no DWA or other message is received by the Net-Net Diameter Director before the *watchdog-timer* value expires, the socket is torn down.

See "[Diameter Heartbeat \(DWR/DWA\)](#)".

Configuring Diameter Director Agents

To configure a Diameter Director Agent:

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter Director agent. The Diameter Director agent table appears in the content area.
3. Click **Add**.
4. **Hostname**—This parameter serves two purposes, first it represents the name parameter that will be used to reference this Diameter Director Agent from within the system. Second you can configure it with an FQDN when resolving this agent via DNS. The **IP address** parameter must be left blank to use an FQDN configured here.
5. Click **Apply**.
6. Select the Diameter Director agent from the table and click **Edit**.

7. **IP address**—Enter an IP address where this Diameter Director Agent can send and receive messages.
8. **Port**—You can override the default port (3868) by setting this to a value of your choosing.
9. **Transport protocol**—Retain the default value of TCP or set it to SCTP to indicate the transport-protocol. This Diameter Director Agent will only use the indicated transport protocol.
10. **Connection mode**—Retain the default value of **outbound** or select **inbound** from the drop-down list for the Net-Net Diameter Director to wait for this agent to initiate the Diameter connection.
11. **State**—Select **enabled** from the drop-down list to activate this Diameter Director Agent.
12. **Description**—Enter a description of this Diameter Director Agent.
13. **Realm**—Enter the realm name where this Diameter Director Agent exists.
14. **Watchdog timer**—Retain the default value of 30 seconds or change it to another value in seconds.

*Hostname	pdagent
IP address	100.10.20.30
Port	3868
Transport protocol	TCP ▼
Connection mode	inbound ▼
State	enabled ▼
Description	
Realm	realm1 ▼
Watchdog timer	30

15. Click **Apply**.

Configuring Supported Applications

To configure Supported Applications on this Diameter Director Agent:

1. Scroll to the Diameter Director application table and click **Add**.
2. **Application ID**—Select an Application ID that this Diameter Director Interface supports from the drop-down list.
3. Click **Apply**.
4. Select the Application ID from the table and click **Edit**.
5. **Application type**—Set this to **accounting** or **authentication** depending on the application type.

6. **Vendor ID**—Enter the Vendor ID for this application ID. (Enter 0 for relay).

*Application ID	16777229
Application type	accounting
Vendor ID	10415

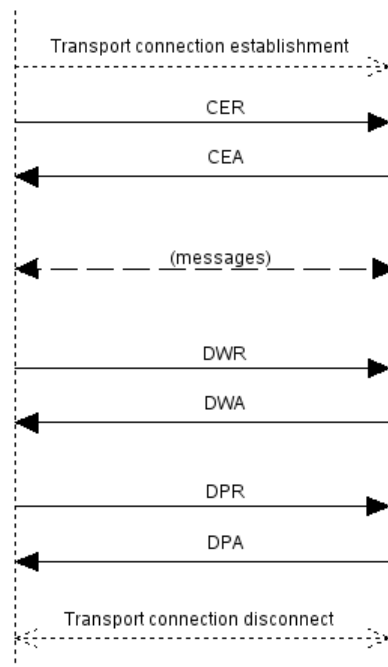
7. Click **Apply**.

Capabilities Exchange Messaging

The establishment of a new Diameter connection between the Net-Net Diameter Director and other Diameter agents is called the Diameter Capability Exchange as per RFC 3588. Capability exchanges facilitate the discovery of a peer agent's identity and its capabilities (protocol version number, supported Diameter applications, security mechanisms, etc.).

Functionally, there are two Capability Exchange scenarios. The first scenario is called the *Ingress Capabilities Exchange* and is when a first-hop Diameter agent discovers the Net-Net Diameter Director by sending it a CER message, and the Net-Net Diameter Director responds with a CEA.

The second scenario is called the *Egress Capabilities Exchange* and is when the Net-Net Diameter Director sends a CER message to its peer and expects a CEA in response.



Ingress Capabilities Exchange

A remote Diameter peer initiates a connection to the Net-Net Diameter Director's Diameter Director Interface listening socket in the Ingress Capabilities exchange. Once the connection is established, the Net-Net Diameter Director starts a 30 second timer while it waits to receive the CER message from the peer. If the timer

expires before the CER is received, the Net-Net Diameter Director closes the transport layer connection.

If the Net-Net Diameter Director receives the CER before the 30 seconds elapses, it begins processing the message. If the received message is not a CER, then the Net-Net Diameter Director closes the connection. Upon receiving a valid CER, the Net-Net Diameter Director starts its work by extracting the Application IDs.

The Net-Net Diameter Director compares the CER's Application IDs to those configured on the *Diameter Director Interface* > *diameter director applications* sub element where the message was received.

If no common application values are found, then the Net-Net Diameter Director sends a CEA message to the peer with Result-Code AVP set to `DIAMETER_NO_COMMON_APPLICATION` and the connection is closed.

When the Net-Net Diameter Director confirms that the peer's CER includes one or more common Application IDs, the Diameter connection is established. The Net-Net Diameter Director returns a CEA message including the common application IDs between CER and those configured on the system. The CEA message includes a Result-Code AVP with value of `DIAMETER_SUCCESS`.

Common Application IDs are maintained with this connection. If a subsequently received message does not include a stored Application ID, the Net-Net Diameter Director replies with a message containing a Result-Code AVP (268) of `DIAMETER_APPLICATION_UNSUPPORTED - 3007`. The answer message also contains Supported-Vendor-Id AVPs whose values are obtained from the *Diameter Director Interface* configuration.

Egress Capabilities Exchange

In the Egress Capabilities Exchange, the Net-Net Diameter Director initiates the connection to a remote peer defined by the *ip address* and *port* parameters in the *diameter director agent* configuration element.

Once a transport-layer connection has been established, the Net-Net Diameter Director sends a CER to the peer. The CER message contains all of the Application-Id AVPs specified in the *diameter director application* configuration elements. If 3 *diameter director application* configuration elements are configured, then all 3 will be sent to the Diameter agent.

If no *diameter director applications* are configured, the Net-Net Diameter Director uses the applications configured for the associated *diameter director group*. If there is no associated Diameter Director Group or if the Diameter Director Group itself has no applications configured, then the applications configured in the Diameter Director Interface are sent in the CER. The set of applications sent in the CER message will be cached. The CER message also contains Supported-Vendor-Id AVPs whose values are configured in the Diameter Director Interface. If the *supported-vendor-ids* list in the PDI configuration is empty, then no Supported-Vendor-Id AVPs will be sent in the CER.

After sending the initial CER, the Net-Net Diameter Director waits 30 seconds to receive the CEA message. This is known as message receive timer - T1. If T1 expires while waiting for the CEA, the Net-Net Diameter Director closes the connection to the Diameter agent.

If a message is received before T1 timer expires, the Net-Net Diameter Director begins parsing and processing the Diameter message. As per RFC 3588, any Diameter message other than a CEA is received during the capability exchange, the

Net-Net Diameter Director closes the connection; any other Diameter message at this point is unacceptable.

Diameter Heartbeat (DWR/DWA)

Device-Watchdog-Request (DWR) and Device-Watchdog-Answer (DWA) messages are used to detect transport failures at the application layer between the Net-Net Diameter Director and a Diameter agent. The request/answer message pair forms a heartbeat mechanism that can indicate if the answering side is not reachable.

Diameter Director Interfaces always respond to a DWR by replying with a DWA message. Diameter Director Interfaces do not initiate DWRs (logically, only agents configured as Diameter Director Agents can initiate DWRs). The Origin-Host AVP (264) and the Origin-Realm AVP (296) in the DWA issued from the Net-Net Diameter Director will be values supplied in the Origin-Host AVP (264) and Origin-Realm AVP (296) in the CEA message.

The Net-Net Diameter Director establishes a DWR/DWA heartbeat between itself and configured Diameter Director Agents. The heartbeat will timeout if the Net-Net Diameter Director does not receive either a Diameter request from the Diameter Director Agent or a DWA in the timeout period. The default value is 30 seconds as suggested by Authentication, Authorization and Accounting (AAA) Transport Profile RFC 2539. The timeout value can also be configured by setting the *watchdog timer* parameter in the *diameter director agent*. Setting the watchdog timer to 0 disables the heartbeat mechanism.

If the watchdog timer expires, the PDA sends DWR to the remote agent. If no DWA is received or other message is received on that connection from the remote agent inside the watchdog timer value, the socket is torn down and a CER is sent toward the agent to reestablish the connection.

Upon detection of a transport failure during the capabilities exchange process, DWRs will not be sent to an alternate peer.

Disconnect Peer Messaging

The Net-Net Diameter Director supports Disconnect Peer Messaging DPR alerts to disconnect down to the transport layer. When the Net-Net Diameter Director receives a DPR, it considers the request and the type of socket where the message was received.

If the DPR was received on a forked socket; a listening socket defined by the *diameter director interface*, the Net-Net Diameter Director replies with a success code in the DPA and closes the socket. The Diameter Director Interface no longer responds to or forwards messages on this socket. A new Diameter connection must be created again with the specific peer that sent the DPR.

If the DPR was received on a Net-Net Diameter Director-initiated connection to a configured Diameter Director Agent, the Net-Net Diameter Director considers the Disconnect-Cause-AVP (273) value. If it is set to REBOOTING (0) or BUSY (1), the Net-Net Diameter Director sends a DPA indicating success and sets the connection out of service. The Net-Net Diameter Director then waits for the peer to tear down the socket. After successful socket teardown, the Net-Net Diameter Director attempts to re-establish a connection to the peer.

If the Disconnect-Cause-AVP (273) is set to DO_NOT_WANT_TO_TALK_TO_YOU (2), the PD sends a DPA indicating success and sets the connection out of service. The PD will then wait for the peer to tear down the socket as defined in section 5.4 of RFC 3588 Diameter Base Protocol. Upon

successful teardown of the socket under these conditions the PD will not try and re-establish a connection to this peer.

Message Type Verification

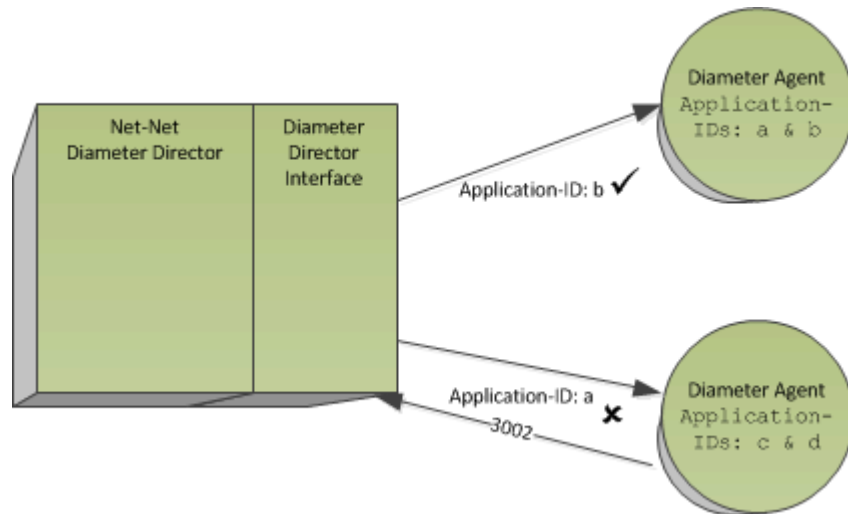
The Net-Net Diameter Director accepts all known and unknown requests, regardless of the Application suite they belong to. The Net-Net Diameter Director attempts to proxy the messages based on the next hop that the Net-Net Diameter Director routing engine determines. Request messages are not filtered (or rejected) if they are not supported by the message's Application ID.

Relay Mode

The Net-Net Diameter Director can operate a Diameter Director Interface in relay mode. This means that it is open to accept and proxy messages of any Application ID value. To run a Diameter Director Interface in relay mode, set the *application-id* value to `0xffffffff` and the *vendor-id* to `0`. Relay mode is the only mode that will take a hex value for the application ID.

If a Diameter agent connects to a Diameter Director Interface running in Relay mode and the agent advertises that it can support 2 Application IDs in its CER/CEA, the Diameter Director Interface will set that connection to use the two Application IDs advertised. After the CER/CEA negotiation, the Net-Net Diameter Director will still accept Diameter messages of any Application type. It will only send messages of the 2 Application IDs to that agent.

If a different Diameter agent connects to the same Diameter Director Interface and advertises 2 additional application IDs, the Net-Net Diameter Director will accept both since it is configured as a relay. Again, all outbound messages to that Diameter message must match either of the two advertised Application IDs.



Routing Diameter Messages

Routing

The Net-Net Diameter Director's routing engine routes incoming Diameter messages to external Diameter agents. The routing engine uses Diameter Director Policies as an evaluation device to determine a routable next hop. The next hop is either a configured Diameter Director Agent or Diameter Director Group.

Conventions in this Chapter

Diameter Director Policy—A container for individual Policy Attributes.

Policy Attribute—Individual evaluations that comprise a Diameter Director Policy.

Diameter Message Reception

First, the Net-Net Diameter Director receives a Diameter message on a Diameter Director Interface. Implicitly, the received message defines the following routing criteria:

- Incoming Realm, interface, address, and port
- Application ID in the Diameter header

The Net-Net Diameter Director starts routing evaluation by considering the message with respect to the Diameter Director Policy named in the Diameter Director Interface's *routing-policy* parameter.



Routing Criteria

Messages are generally routed to a next hop based on the contents of one or more AVPs. The tests to determine a match work similarly to the Net-Net SBC's HMR functionality.

The Net-Net Diameter Director uses a Diameter Director Policy to evaluate a Diameter message. The Diameter Director Policy is a container for a group of Policy Attributes which are configured as sub-elements. Policy Attributes define the individual comparisons used for routing decisions. The order that Policy Attributes are tested in is determined by the configured *priority* parameter, lower values evaluated first. If two Policy Attributes have the same configured priority, the one which was configured first is evaluated first.

Once the first Policy Attribute has been identified, the routing engine evaluates the message against that Policy Attribute.

The match operation is configured by defining a *match field* and *match value*. The Net-Net Diameter Director evaluates if the *match value* is found in the match field.

The *match field* is usually an AVP in a Diameter message configured as `avp=<avp-number>`.

For example, if the match field is configured as

`avp=264`

and the match value is configured as

`.com`

Then the routing engine will make a positive match in the Origin-Host (264) AVP with the string or partial string of ".com" (Given that the AVP Type and Comparison type parameter are correctly set: see next section).

Routing Evaluation Data Types

The Net-Net Diameter Director must parse and evaluate the match field correctly to produce an accurate match. Otherwise, the wrong comparison might be made.

AVP Type

The *avp type* of the *match field* is defined so that the Net-Net Diameter Director knows the encoding of the data it is evaluating. AVP types can be found in either the respective RFC or from context. Valid *avp type* values are:

- Octet string
- Diameter URI
- Integer32
- Unsigned Integer 32
- Diameter Enumerated
- Diameter Address
- Grouped

Comparison Type

The *comparison type* is defined so that the Net-Net Diameter Director knows the correct way to determine if the *match value* appears in the *match field*. Valid *comparison types* are:

- Integer—literal number match
- Regex—regular expression match
- Case-sensitive—case-sensitive string match
- Case-insensitive—case-insensitive string match
- Refer-case-sensitive—case sensitive when referring to a previous Diameter Director Policy, defined by \$diameter-director-policy
- Refer-case-insensitive—case insensitive when referring to a Diameter Director Policy, defined by \$diameter-director-policy
- grouped-avp—matching on data found within a grouped AVP

Reserved Values

A list of five reserved values can be used in the match value parameter. These values are:

- \$avp=<avp-num>—AVP number
- \$appln_id—application ID of the request being received

- `$cmd_code`—command code
- `$inc_realm`—incoming realm
- `$inc_intf`—incoming interface

Routing Decision Process

The evaluation of match value against match field can produce either a positive match (match value found in match field) or a failed match (match value NOT found in match field).

The Net-Net Diameter Director evaluates each Policy Attribute in turn according to its priority. If all Policy Attributes in a Diameter Director Policy are tested without any making a match, then the Net-Net Diameter Director sends a message to the originating host with DIAMETER_UNABLE_TO_DELIVER-3002 code in a Result-Code AVP (268).

When the routing engine makes its first positive match on a Policy Attribute, it checks if the *next policy* parameter is configured. If it is configured with a valid Diameter Director Policy, the Net-Net Diameter Director starts the evaluation process again using the original Diameter message against the newly referenced Diameter Director Policy.

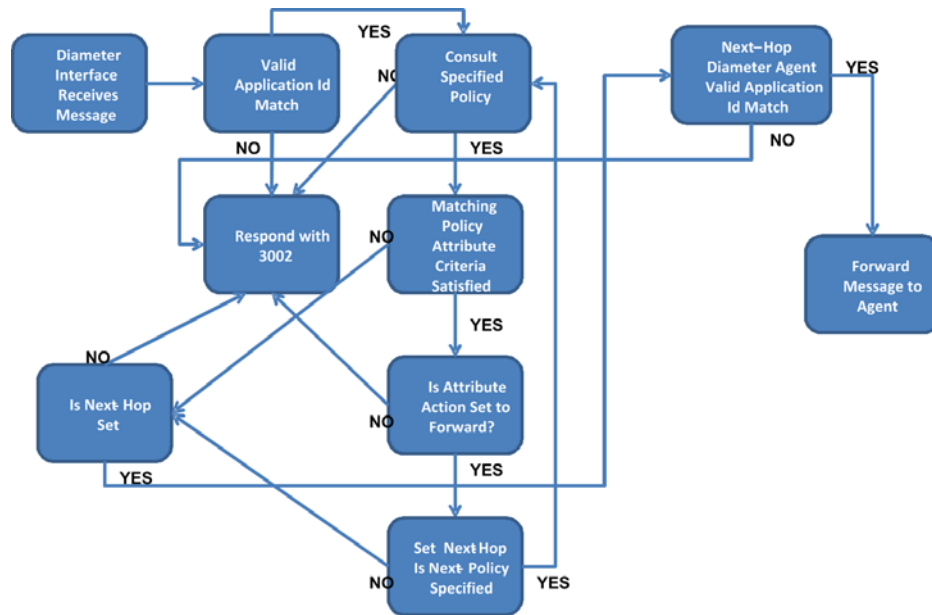
The routing engine continues, following the chain of Policy Attributes within Diameter Director Policies until it makes a match on a Policy Attribute with an empty *next policy* parameter, i.e. a blank value.

If there is no configured *next policy*, the routing engine looks at the *action* parameter. If the *action* is set to reject, then the Net-Net Diameter Director sends a message back to the originating host with DIAMETER_UNABLE_TO_DELIVER-3002 code in a Result-Code AVP (268). If the value is set to *forward*, the Net-Net Diameter Director performs an Application ID check. This check insures that the target next hop had previously stated it could support the ApplicationID found in the Diameter message being routed. This was performed in the CER/CEA transaction. If the Diameter agent can support the ApplicationID, then the Net-Net Diameter Director sends the message to the element defined in the *next hop* parameter.

The next hop is configured as either a configured Diameter Director Agent or Diameter Director Group; it cannot be configured as an IP address. The *next hop* value must be the Diameter Director Agent or Diameter Director Group's configured *name* parameter. Diameter Director Group groups are configured as `ddg:<diameter-director-group-name>`.

If no next hop is configured, the Net-Net PD drops the request message and sends a DIAMETER_UNABLE_TO_DELIVER-3002 code in a Result-Code AVP (268) to the requester.

The following diagram represents the routing decision process graphically:



Record Route AVP Creation

When a Diameter message is forwarded to a Diameter agent, the Net-Net Diameter Director inserts the Route-Record AVP (282) into the message. The value in this AVP is the one that the Net-Net PD inserted into the Origin-Host AVP (264) during the CER/CEA negotiation with that target Diameter agent.

Grouped AVP Routing

The Net-Net Diameter Director can route messages based on the data contained in Grouped AVPs. RFC 3588 states that a Grouped AVP is specified as a sequence of AVPs, including their headers and padding. Grouped AVP handling is configured within a Policy Attribute.

The Net-Net Diameter Director is configured to consider a set of AVPs as a Grouped AVP, which model the definition of that grouped AVP. One or more member AVPs of the group are parsed to find if a *match value* is found in the *match field*. All configured *sub avps* must make positive matches to consider the whole Policy Attribute as having made a positive match. That is, the comparison is an AND function for all *sub avps* that comprise the grouped AVP.

To configure a Grouped AVP-handling Policy Attribute, first set the *avp type* to grouped and the *comparison type* to grouped-avp. The next step is to configure sub avps. These act like Policy Attributes, although they can only be configured to match on an AVP. Not all contents of the grouped AVP must be specified as *sub avps*, but all of those that are configured as *sub avps* must match for that Policy Attribute to return a positive match.

Sub avps are configured similarly to Policy Attributes. The *avp code* parameter indicates the member AVP where the match is being tested; you can only match on AVP data in a *sub avp*. The remaining [AVP Type](#), [Comparison Type](#), and *match value* parameters represent the same data as they do in the Policy Attribute configuration.

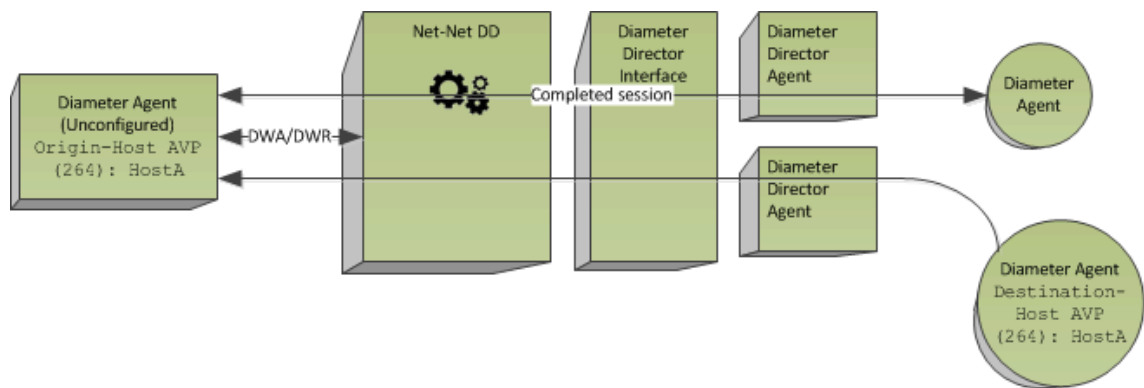
Dynamic Routing

The Net-Net Diameter Director can route a request message from a Diameter Director Agent to a non-configured Diameter agent if that non-configured Diameter agent had already sent a request through the Net-Net Diameter Director.

To contact non-configured agents, the Net-Net Diameter Director caches the non-configured agent's return route for as long as the connection between itself and the endpoint remains up. The Net-Net Diameter Director uses the value of the Origin-Host AVP (264) sent from the non-configured endpoint as the key to reference the route.

Subsequently, when a configured Diameter Director Agent wants to initiate a request to the non-configured agent, the Net-Net Diameter Director can do so without having explicit configuration to support it using the cached route.

Consider the following diagram: Since the route to HostA is already cached, the Net-Net Diameter Director will route any message addressed to Host A when the Diameter agent uses the correct value in the Destination-Host AVP.



When the unconfigured Diameter agent's connection is lost, route entries using that connection are no longer valid and therefore will be removed from all the sockets. The Net-Net Diameter Director then has no route to reach that target.

Policy Rejection

When the Net-Net Diameter Director reaches a Policy Attribute with no next-hop and whose *action* is set to **reject**, a configurable result code can be sent to the originating Diameter agent.

By default, the Net-Net Diameter Director sends an error response with Result-Code AVP set to 3002 (UNABLE_TO_DELIVER) to the requesting Diameter agent. Alternatively the Net-Net Diameter Director can send a configurable Result-Code AVP or Experimental-Result-Code AVP.

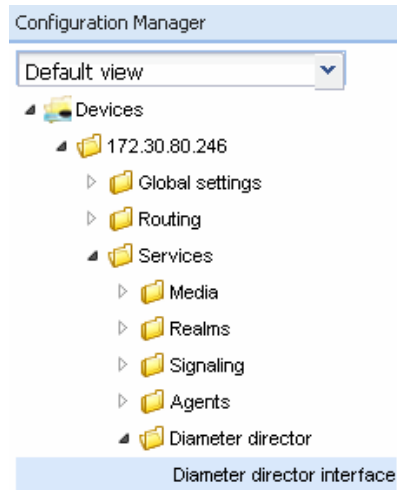
To configure the value of the Result-Code AVP included in the error message, set the *reject-result-code* parameter in the Policy Attribute configuration element. To configure the value of the Experimental-Result-Code AVP included in the error message, set the *reject-exp-result-code* and *reject-exp-vendor-id* parameters in the Policy Attribute configuration element.

Setting the *reject result code* parameter to 0 uses the default Result-Code AVP on a policy reject. Setting either the *reject-exp-result-code* or *reject-exp-vendor-id* parameter to 0 does not send an Experimental Result Code AVP on a policy rejection.

Configuring Root Diameter Director Policies

To configure the root Diameter Director Policy on a Diameter Director Interface:

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter Director interface. The Diameter Director interface table appears in the content area.
3. Select a Diameter Director interface from the table and click **Edit**.
4. **Routing policy**—Select the name of root Diameter Director policy to first be applied to incoming messages on this Diameter Director interface from the drop-down list.

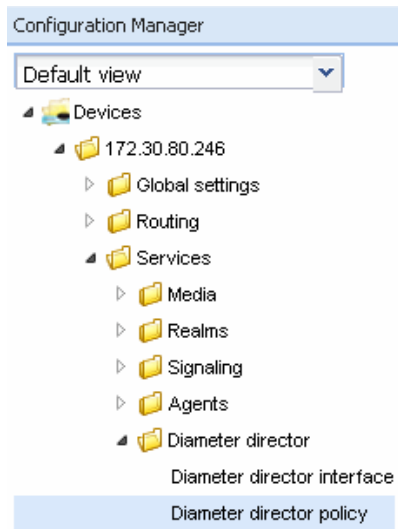


5. Click **Apply**.

Diameter Director Policy

To configure the Diameter Director Policy:

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter Director policy. The Diameter Director policy table appears in the content area.
3. Click **Add**.
4. **Name**—Enter the name of this Diameter Director policy. This name will be referenced from Policy Attributes or from a Diameter Director interface.
5. Click **Apply**. The Diameter Director policy now appears in the table.
6. Select the policy from the table and click **Edit**.
7. **State**—Select **enabled** from the drop-down list to use this Diameter Director policy.

Policy Attributes

1. Scroll to the Diameter attribute table and click Add.
2. **Attribute name**—Enter the name of this policy attribute.
3. **Comparison type**—Select the type of computational comparison the Net-Net Diameter Director uses to test for a Diameter Director policy match from the drop-down list. Valid values are:

regex | integer | case-sensitive | case-insensitive | refer-case-sensitive | refer-case-insensitive | grouped-avp

4. **Match value**—Enter the value within the match-field to find and make a positive match on.
5. **Match field**—Enter the field within the Diameter message to search for the match-value in. This is required. Valid values are:

Set this to **avp=<avp-num>** to match on a specific AVP field

command-code—Used to make match on the Diameter command code

application-id—Used to make a match on an application id in the Diameter message

incoming-realm—Used to make a match on the realm where this message was received

incoming-interface—Used to make a match on the network interface where this message was received

*Attribute Name	name
*Comparison type	case-sensitive
*Match value	ab
*Match field	command-code

- Click **Apply**.
- Select the Policy attribute from the table and click **Edit**.
- Action**—Select either **forward** or **reject** from the drop-down list as the action to take after making a positive match on the previously entered match-value.

Action	forward - reject - forward - none

- AVP type**—Select the data type of the content of the match field from the drop-down list. Refer to the Diameter standards document for the encodings of individual AVPs. Valid values are:

octet-string | integer32 | unsignedint32 | address | time | utfstring |
diameterident | diameteruri | enumerated | grouped

AVP type	none - diameteruri - time - enumerated - address - integer32 - grouped - none - octet-string - unsignedint32

- Priority**—Enter the priority in which to execute this policy attribute with respect to the other policy attributes configured in this Diameter Director Policy. The lowest priority policy attribute is executed first.
- Reject result code**—Enter the value to include in the Result Code AVP when the Net-Net Diameter Director chooses this Policy Attribute with a **reject** action.
- Reject experimental result-code**—Enter the value to include in the Experimental Result Code AVP when the Net-Net Diameter Director chooses this Policy Attribute with a **reject** action. This parameter must be configured along with the *Reject experimental vendor ID* parameter.
- Reject experimental vendor ID**—Enter the vendor ID to accompany the Experimental Result Code when the Net-Net Diameter Director chooses this Policy Attribute with a **reject** action. This parameter must be configured along with the *Reject experimental result code* parameter.

14. **Next hop**—Enter the next hop, where to forward the Diameter message when all other policy matching criteria are met. This value is the *Hostname* parameter of a Diameter Director Agent. To indicate a Diameter Director group, use the prefix **pdg**:
15. **Next policy**—Select the Diameter Director Policy *Name* to use against the Diameter message when all other policy matching criteria are met from the drop-down list.

*Comparison type	case-sensitive ▼
Priority	0
Reject result code	3002
Reject experimental result code	0
Reject experimental vendor ID	0

16. Click **Apply**.

Policy Attributes for Grouped AVP

1. Scroll to the Sub AVP table and click **Add**.
AVP type must be set to **grouped** and *Comparison type* must be set to **grouped-avp** to use this feature.
2. **AVP code**—Enter the grouped AVP code number to make a match within.
3. Click **Apply**.
4. Select the AVP code from the table and click **Edit**.
5. **AVP type**—Select the data type from the drop-down list of the content of the AVP data the Net-Net Diameter Director is parsing to make a match within the AVP named in the avp-code. Refer to the Diameter standards document for the encodings of individual AVPs. Valid values are:
octet-string | integer32 | unsignedint32 | address | time | utfstring | diameterident | diameteruri | enumerated
6. **Comparison type**—Select the type of computational comparison from the drop-down list the Net-Net Diameter Director uses to test for a Diameter Director policy match. Valid values are:
regex | integer | case-sensitive | case-insensitive | refer-case-sensitive | refer-case-insensitive
7. **Match value**—Enter the explicit value within the AVP defined in the avp-code parameter of this configuration element to find and make a positive match on.
8. Click **Apply**.

Diameter Director Group Recursive Routing

When a Diameter Director Group is selected as a *next hop* by the Policy Attribute, the group determines an agent to route the request to based on *strategy*. Upon receiving an error response from that agent in the group, the response is proxied back to the originator of the request.

Recursive routing is when the Net-Net Diameter Director can resend the Diameter message to the next available Diameter Director Agent in a Diameter Director Group when it receives a defined error message. This behavior is used when the *do recursion* parameter is **enabled** in the recursive routing subelement under the following conditions:

- The error message matches a value configured in the *result codes* parameter in the *recursive routing* subelement.
- The error message matches a value configured in the *exp result codes* parameter in the *recursive routing* subelement (for errors contained in Experimental Result Code AVP).
- A response is not received within the value in milliseconds configured in the *transaction timeout* parameter in the *recursive routing* subelement. This timer is started when a request is sent to an agent.

Further, the recursion to the next agent is ended in the following cases:

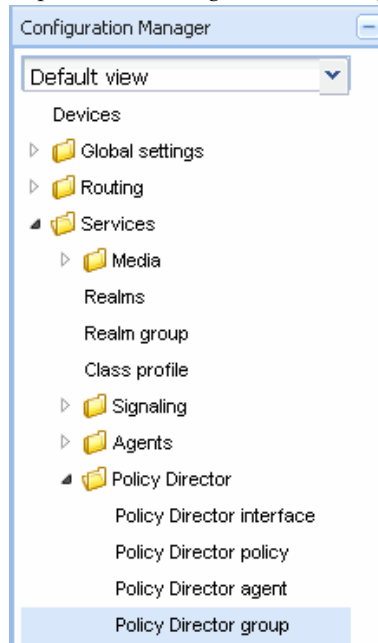
- The result code in the response does not match any configured values (*result codes* nor *exp result codes*).
- The *recursion timeout* (milliseconds) parameter in the *recursive routing* subelement expires while performing the recursion before a valid final response is received. This timer starts when the Net-Net Diameter Director sends the request to the first agent in the Diameter Director Group.
- An error response is returned for the last agent unused agent in the Diameter Director Group.
- All remaining agents in the Diameter Director Group are unavailable (Out-Of-Service).

If no more Diameter Director Agent are available to retry sending a message to, then a 3002 (UNABLE_TO_DELIVER) response is sent back to the originating requester.

Configuring Recursive Routing

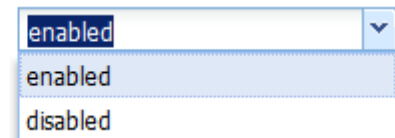
To configure recursive routing on a Diameter Director Group:

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter Director group. The Diameter Director group table appears in the content area.
3. Select the Diameter Director group you want to configure from the table and click **Edit**.
4. Scroll to the Recursive routing table.
5. Select a Recursion from the table and click **Edit**.
6. **Recursion**—Select enabled from the drop-down list to use recursive routing for Diameter Director Groups.

*Recursion



7. **Result codes**—Click **Add** to enter the Result-Code AVP values, as returned by an agent, which prompts the Net-Net Diameter Director to recurse through this Diameter Director Group.

Result codes
2200

8. Click **OK**.
9. Repeat steps 7-8 to add more Result codes.
10. **Experimental result codes**—Click **Add** to enter Experimental result code AVP values, as returned by an agent, which prompts the Net-Net Diameter Director to recurse through this Diameter Director Group.

Experimental result codes
5300

11. Click **OK**.
12. Repeat steps 10-11 to add additional Experimental result codes.

13. **Transaction timeout**—Enter the time in milliseconds which the Net-Net Diameter Director can take to recurse the current Diameter Director Agent in this Diameter Director Group before failing and trying the next Diameter Director Agent in the group or returning a 3002 to the requesting agent.
14. **Recursion timeout**—Enter the total time in milliseconds which the Net-Net Diameter Director can take to recurse through all Diameter Director Agents in this Diameter Director Group before failing and returning a 3002 to the requesting agent.

*Transaction timeout (msec)	32000
*Recursion timeout (msec)	100000

15. Click Apply.

Load Balancing & Redundancy

Diameter Director Groups

A Diameter Director Group combines several Diameter Director Agents into a single logical entity. Each configured Diameter Director Agent may only belong to one Diameter Director Group.

The Diameter Director Group is defined by the *name* parameter. You can also provide a *description*. All Diameter Director Agents that comprise the Diameter Director Group are added to the *destinations* parameter. The Diameter Director Group can then be a next-hop in a Diameter Director Policy. This is configured by prepending the Diameter Director Group name with a `ddg:` prefix. For example: `ddg:exampleddg`

When the next-hop is a Diameter Director Group and if one of the Diameter Director Agents in the group becomes unavailable, the Net-Net Diameter Director can still communicate with the peer via an alternate Diameter Director Agent. Where the Net-Net Diameter Director communicates with multiple peers, the Diameter Director Group also provides the ability to distribute the traffic among them depending on the message distribution strategy.

Diameter Director Group Strategy

When the Net-Net Diameter Director settles on sending a Diameter message to a *next hop* via a Diameter Director Policy lookup, the *strategy* parameter in the *diameter-director-group* indicates how to choose the discrete Diameter Director Agent in that group. Diameter Director Agent selection is performed on a per-message basis. The two strategies for Diameter Director Agent selection are

- **Hunt**—The Diameter Director Group selects the Diameter Director Agent in the order they were configured. If the first agent is in service, then it is selected. If the first agent is out of service, then the second or the next available Diameter Director Agent is selected. As long as the first agent remains in service, all incoming requests to the Diameter Director Group are forwarded there.

- Round Robin—The Diameter Director Group distributes incoming requests among all Diameter Director Agents in the order they were configured. The Diameter Director Agent selection begins with the first configured agent which is in service. Upon receiving the next request, the next Diameter Director Agent is selected and so on. This strategy distributes incoming requests among all Diameter Director Agents uniformly that comprise the Diameter Director Group.

Diameter Director Group States

A Diameter Director Group's Diameter Director Agent members exist in one of the two states: In Service or Out Of Service. At startup, Diameter Director Agents are considered OOS until the Net-Net Diameter Director establishes a successful Diameter connection with the configured Diameter Director Agent. The Diameter connection is defined as a valid transport layer connection followed by a successful Capability exchange causing the connection to be kept open. Once In Service, the Diameter Director Agent can return to Out of Service if the Diameter connection is lost.

When the Net-Net Diameter Director selects a Diameter Director Agent from a Diameter Director Group as the destination for a Diameter message, if it is Out of Service it is never chosen. Once a Diameter connection is re-established, the Diameter Director Agent is considered In Service again and can participate in the Diameter Director Group selection process.

Diameter Director Group startup and shutdown

When the Net-Net Diameter Director's Diameter Director Group configuration is modified (or removed completely) and then reloaded, the connections from the Net-Net Diameter Director to the Diameter Director Agents in that group are all closed and renegotiated from the CER/CEA forward.

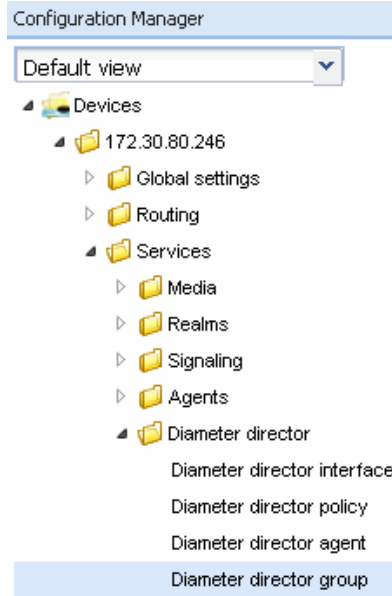
Any application IDs from the Diameter Director Group configuration will be used for negotiation for all Diameter Director Agent, unless the Diameter Director Agent has its own configured application ID.

To initiate the process of closing and opening connections to Diameter Director Agents, the Net-Net Diameter Director sends DPRs to all affected agents. After the DPA is received, the transport layer connection is reset. If the Net-Net Diameter Director does not receive a DPA within 32 seconds of sending the DPR, the peer is deemed unresponsive and the connection is closed.

Configuring Diameter Director Groups

To configure a Diameter Director Group:

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter Director group. The Diameter Director group table appears in the content area.
3. Click **Add**.
4. **Group name**—Enter the name of this Diameter Director group. The value entered here will be referenced from a Diameter Director Policy Next hop parameter.
5. **Recursion timeout**—Enter the total time in milliseconds which the Net-Net Diameter Director can take to recurse through all Diameter Director Agents in this Diameter Director Group before failing and returning a 3002 to the requesting agent.
6. **Recursion**—Select **enabled** from the drop-down list to use recursive routing for Diameter Director Groups.
7. **Transaction timeout**—Enter the time in milliseconds which the Net-Net Diameter Director can take to recurse the current Diameter Director Agent in this Diameter Director Group before failing and trying the next Policy Direct Agent in the group or returning a 3002 to the requesting agent.
8. Click **Apply**.
9. Select the Diameter Director group from the table and click **Edit**.
10. **Description**—Enter a description of this Diameter Director group.
11. **State**—Select **enabled** from the drop-down list to use this Diameter Director group.
12. **Strategy**—Select either **hunt** or **round-robin** from the drop-down list to indicate which way a Diameter Director Agent is chosen for sending messages to.
13. **Destinations**—Click **Add**.
14. **Destination**—Select the *hostname* value of a Diameter Director agent from the drop-down list to add that agent to this Policy Director Group.

15. Click **Apply**.

Diameter Message Manipulations

Manipulation Application

The Net-Net Diameter Director can perform manipulations on all non-grouped AVPs. A message manipulation is the ability to search for a predefined string via within an AVP and then replace it with another value. This is similar to the Net-Net SBC's header manipulation rules functionality.

Diameter Manipulations can be applied at one of three logical points in the configuration as the message is received or forwarded: Diameter Director Interface, Diameter Director Group, Diameter Director Agent. The Diameter Director Interface has the highest priority, the Diameter Director Group (if configured) has the second highest priority, and Diameter Director Interface has the lowest priority. If more than one manipulation is configured for the trip from Diameter Director Interface to Diameter Director Agent, the manipulation with the highest priority is executed.

A *diameter manipulation* configuration element is defined by a *name* parameter. You can optionally add a *description* field to the diameter manipulation. Within each diameter manipulation you can configure multiple *manipulation rule* subelements. The manipulation rule subelements are the configuration where AVPs are identified, searched, and in which the data is replaced.

Manipulation Rule

Creating a manipulation rule is divided into three parts, defining the message type and AVP where the manipulation is performed, defining how the search on the AVP is performed, and defining what to replace the found string with.

You must first define the *name* parameter of the manipulation rule. Optionally you can add a *descr avp code* parameter that is a description of this manipulation rule.

Defining the AVP to Search

The Diameter message command code is defined by configuring the *msg cmd code* parameter with a numeric value. In addition, the message type as **request**, **response**, or **all** must be configured in the *msg type* parameter.

After the Net-Net Diameter Director knows where to look for the AVP, the *avp code* must be defined with the numeric value of the AVP to be searched. Also the AVP data type is defined so Net-Net Diameter Director knows how to correctly parse the AVP once found. This is configured in the *avp type* parameter with valid values of: octet-string, integer32, unsignedint32, address, time, utfstring, diameterident, diameteruri, or enumerated.

Defining the Value Search

The *comparison type* is defined so that the Net-Net Diameter Director knows the correct way to determine if the *match value* appears in the *avp code*. Valid *comparison types* are:

- Integer—literal number match
- Regex—regular expression match
- Case-sensitive—case-sensitive string match
- Case-insensitive—case-insensitive string match
- Refer-case-sensitive—case sensitive when referring to a previous Diameter Director policy, defined by \$diameter-director-policy
- Refer-case-insensitive—case insensitive when referring to a Diameter Director policy, defined by \$diameter-director-policy

Finally, the match operation is configured by defining a *match value*, which is the string to find. The Net-Net Diameter Director evaluates if the *match value* is found in the *avp code* AVP.

New Value to Substitute in the AVP

When the *match-value* string is found, the Net-Net Diameter Director references the *action* parameter. This is configured as either **add**, **replace**, or **remove** and indicates the action to perform on the string.

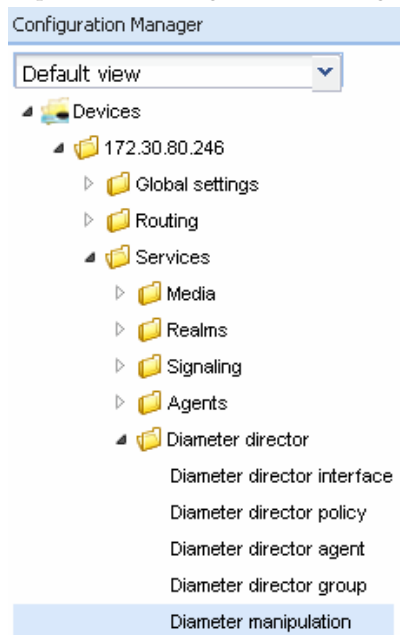
If the action is set to **add** or **replace**, the value configured in the *new value* parameter is added in place of (replace) or after (add) the found match value. Similarly, if *action* is set to **remove**, the found string is removed from the AVP.

If the match-value is not found, the Net-Net Diameter Director continues processing the message without any AVP manipulation.

Configuring Diameter Manipulation

To configure a diameter manipulation configuration element:

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter manipulation.
3. The Diameter manipulation table displays in the content area.
4. Click **Add**. The Add Diameter manipulation dialog box displays.

5. **Name**—Enter the name of this Diameter manipulation element.
6. Click **Apply**.
7. Select the Diameter manipulation element you want to configure in the table and click **Edit**.
8. **Description**—Enter an optional description for this Diameter manipulation element.
9. Click **Apply**.

Configuring Diameter Manipulation Rules

To configure Diameter

1. Scroll to the Diameter manipulation rule table.
2. Click **Add**.
3. **Name**—Enter the name of this manipulation rule.
4. **AVP code**—Enter the AVP code to manipulate.

*Name	maniprule1
Sequence number	1
*AVP code	2200

5. Click **Apply**.
6. Select the Diameter manipulation rule from the table and click **Edit**.
7. **AVP type**—Select the data type from the drop-down list of the content of the match field. Refer to the Diameter standards document for the encodings of individual AVPs. Valid values are:
 octet-string | integer32 | unsignedint32 | address | time | utfstring | diameterident | diameteruri | enumerated | grouped | comparison-type
8. **Action**—Select either **add**, **replace**, or **remove** from the drop-down list as the action to take after making a positive match on the previously entered match-value.
9. **Message type**—Set this to the type of message this manipulation applies to as **request**, **response**, or **all**.
10. **Message command code**—Select the command code number of the message to execute the manipulation on from the drop-down list.
11. **Comparison type**—Select the type of computational comparison the Net-Net Diameter Director uses to test for a Diameter Director policy match from the drop-down list. Valid values are:
 regex | integer | case-sensitive | case-insensitive | refer-case-sensitive | refer-case-insensitive | grouped-avp
12. **Match value**—Enter the value within the match-field to find and make a positive match on.

13. **New value**—Enter the value that should be added or replaced in the old match-value's position.

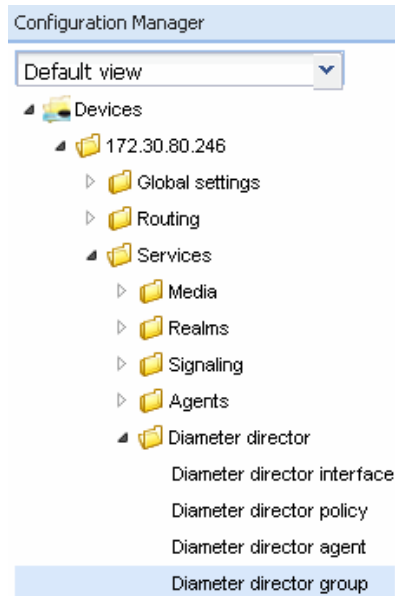
AVP type	address
Action	add
Message type	request
Message command code	274
Comparison type	case-sensitive

14. Click **Apply**.

Applying the Manipulation

You can apply a diameter manipulation by name to either the Diameter Director agent, Diameter Director interface or Diameter Director group. All three configuration elements contain the same two parameters: in-manip-id, out-manip-id

1. Expand the Configuration Manager slider > Services > Diameter Director.



2. Click Diameter Director group. The Diameter Director group table appears in the content area.
3. Select the Diameter Director group you want to configure from the table and click **Edit**.
4. **Inbound manipulation**—Select the name of an existing diameter manipulation from the drop-down list to apply as received by the Net-Net Diameter Director on this element.
5. **Outbound manipulation**—Select the name of an existing diameter manipulation from the drop-down list to apply as forwarded from the Net-Net Diameter Director on this element.
6. Click **Apply**.

Message Manipulation Caveats

CER/CEA

CER/CEA messages are not proxy-able and any manipulation to these messages may affect the capabilities exchange process. Some of the values for egress CER/CEA are configurable through ACLI, generally these configurable values are not subject to manipulation.

Received CER/CEA Messages

Since the Net-Net Diameter Director has no direct control over the CER/CEA it receives, most AVPs in the ingress case can be manipulated. See the following table. Note that manipulating ingress CER/CEA messages can likely affect the operation of the Net-Net Diameter Director, resulting in rejections.

AVP	CER/CEA	Manipulation Actions	Comments
Origin-Host	Both	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Origin-Realm	Both	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Vendor-ID	Both	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Product-Name	Both	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Origin-State-Id	Both	Add/Delete	See Origin-State-Id AVP
Firmware-Revision	Both	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Error-Response	CEA	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value

Sent CER/CEA Messages

AVPs in the egress CER and CEA messages which derive their values from ACLI configuration may not be manipulated even if the user configures a manipulation rule for them; these rules will be ignored. The following table lists the AVPs that may not be manipulated and their corresponding ACLI configurations:

AVP	CER/CEA	ACLI Configuration	Comments
Origin-Host	Both	diameter-director-interface -> origin-host-identifier diameter-director-interface -> origin-host-format	User can configure the origin-host-identifier and origin-host-format to change the Origin-Host-AVP to reflect the desired value.
Origin-Realm	Both	media-manager->realm-config	User can configure the name of the realm associated with the Diameter Director Interface to reflect the desired value.
Host-IP-Address	Both	diameter-director-interface-> diameter-director-ports->address	This AVP is used to inform the peer of the sender's IP address. So no manipulations to this AVP shall be allowed. The user can change the IP address used in PDI to change this value.
Supported-Vendor-Id	Both	diameter-director-interface->supported-vendor-id	User can remove/add vendor-ids from this list to reflect the desired value(s) for this AVP.
Auth-Application-Id	Both	diameter-director-interface-> diameter-director-applications / diameter-director-group->diameter-director-applications / diameter-director-agent->diameter-director-applications	User can set the: application-type=authentication vendor-id=0 application-id=<id>, to see the correct Auth-Application-Id AVP in the message.
Acct-Application-Id	Both	diameter-director-interface->diameter-director-applications / diameter-director-group->diameter-director-applications / diameter-director-agent->diameter-director-applications	User can set the : application-type=accounting vendor-id=0 application-id=<id>, to see the correct Acct-Application-Id AVP in the message.
Vendor-Specific-Application-Id	Both	diameter-director-interface-> diameter-director-applications / diameter-director-group-> diameter-director-applications / diameter-director-agent-> diameter-director-applications	User can set the : application-type=authentication/accounting vendor-id=<id> application-id=<id>, to see the correct Vendor-Specific-Application-Id AVP in the message.
Result-Code	CEA	N/A	The value of this AVP is determined by the Net-Net Diameter Director based on whether or not it successfully processed the CER request. Any changes to this AVP value in the egress CEA could potentially cause problems in the Net-Net Diameter Director.

The following table lists the AVPs from the CER and CEA message that may be manipulated and the kind of operations allowed on the listed AVP:

AVP	CER/CEA	Default (If any)	Manipulation Actions	Comments
Vendor-Id	Both	1948	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Product-Name	Both	Acme Packet Agent	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Origin-State-Id	Both	None	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Firmware-Revision	Both	None	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value
Error-Message	CEA	None	Add/Replace/Delete	If the AVP already exists in the received message, then the Add action will replace the value with the configured new-value

AVP Manipulation Caveats

Origin-State-Id AVP

If the Origin-State-Id (278) AVP is modified it could cause the connection to be torn down since a change in value could indicate to a peer that the agent communicating to it has lost its state.

Thus Origin-State-Id (278) can only be added or removed from Diameter messaging, you may not set the *action* parameter to replace for manipulating the AVP.

Experimental-Result Code Manipulation

The Net-Net Diameter Director can manipulate the Result-Code AVP (268) in response messages according to expected behavior.

The Net-Net Diameter Director can manipulate the Experimental-Result-Code AVP (298) in response messages in a similar fashion to the Result-Code AVP. The Experimental-Result-Code AVP is part of Experimental-Result grouped AVP as shown below:

```
Experimental-Result ::= < AVP Header: 297 >
                        { Vendor-Id }
                        { Experimental-Result-Code }
```

When the avp-code is set to 293 in the manipulation rules configuration, the manipulation engine parses for Experimental-Result-Code AVP (298) inside the Experimental-Result AVP (297) and manipulates that value based on the configuration. Both inbound and outbound manipulations are supported.

Security

Anonymous Diameter Agent Blocking

The Net-Net Diameter Director can prevent anonymous Diameter agents from connecting to specific Diameter Director Interfaces. The Diameter Director Interface contains the *allow anonymous* parameter in the *diameter director ports* subelement. You can configure the Net-Net Diameter Director to allow all connections to the Diameter Director Interface's IP address or allow only Diameter agents that are formally defined as Diameter Director Agents to connect. The *allow anonymous* parameter defines this behavior by accepting either of the following arguments:

- **all**—Any Diameter agent may connect to this Diameter Director Interface.
- **agents-only**—Only Diameter agents defined as Diameter Director Agents will be allowed by the Net-Net Diameter Director. Any connection attempt from an undefined Diameter peer will be rejected. If an unauthorized Diameter agent connects to the Net-Net Diameter Director, the TCP connection request will be rejected.

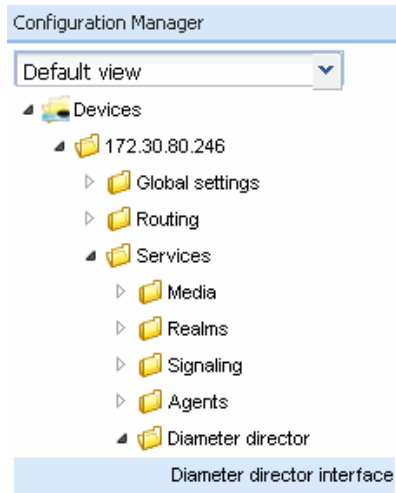
In addition to configuring the Diameter Director Interface, Diameter agents have to be pre-defined. This is similar to defining an ACL for known agents. Once a Diameter agent is defined within the Net-Net Diameter Director configuration, it is *known* so that it can connect to a Diameter Director Interface that is set to **agents-only**.

Normally a Diameter Director Agent configuration element defines the behavior that the Net-Net Diameter Director initiates a Diameter connection to that agent. The Diameter Director Agent is automatically considered authorized for connecting to a Diameter Director Interface. But setting the *connection mode* parameter to **inbound**, the Net-Net Diameter Director waits for the Diameter agent to initiate the connection. This lets agents remain unconnected to the Net-Net Diameter Director, but still remain authorized to connect when needed.

Configuring Diameter Director Interfaces

To configure a Diameter Director Interface to only accept connections from configured Diameter Director Agents:

1. Expand the Configuration Manager slider > Services > Diameter director.
2. Click Diameter director interface.

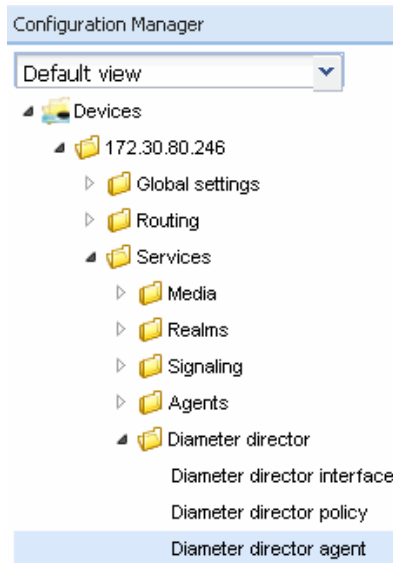


3. Click the Diameter Director interface you want to configure in the Diameter Director interface table and click **Edit**.
4. Click the Diameter Director port you want to configure to accept connections from configured Diameter Director agents and click **Edit**.
5. **Allow anonymous**—Click **agents-only** in the drop-down list for the Net-Net Diameter Director to only accept connections from configured Diameter Director Agents. The default value is **all**.
6. Click **Apply**.

Diameter Director Agent

To configure a Diameter Director Agent as authorized to connect to this Net-Net Diameter Director, yet let the Net-Net Diameter Director wait for that connection:

1. Expand the Configuration Manager slider > Services > Diameter director.
2. Click Diameter director agent.



3. Click the Diameter Director agent hostname you want to configure in the Diameter Director agent table and click **Edit**.
4. **Connection mode**—Click **inbound** in the drop-down list for the Net-Net Diameter Director to wait for this agent to initiate the connection. Alternatively, leaving this parameter as outgoing configures the Net-Net Diameter Director to start a capability exchange negotiation toward this Diameter Director Agent. The default value is **outbound**.
5. Click **Apply**.

Supported Commands for KPI Tracking

The Net-Net Diameter Director supports maintaining Key Performance Indicators (KPI) for Commands that are supported by the following Diameter interfaces:

Interface	Specification	Application-Id	Vendor-ID
Rx	3GPP TS 29.211	16777229	10415
Gq	3GPP TS 29.209	16777222	10415
Rq	ETSI ES 283.026	16777222	13019 10415 ¹
Gy	IETF RFC 4006	4	0
S6a	3GPP TS29.272	1677251	10415
S6d	3GPP TS29.272	1677251	10415

Interface	Specification	Application-Id	Vendor-ID
Gx		16777238	
Gxx		16777266	

¹The vendor identifier assigned by IANA to 3GPP is 10415 and the vendor identifier assigned by IANA to ETSI is 13019. The Vendor-Id header for AVPs imported from TS 129 209 [7] shall be set to 3GPP (19415), while AVPs defined in the present document or imported from TS 183 017 [5] or ES 283 034 [4] shall be set to ETSI (13019)

The list of commands found in the previously listed Interfaces is as follows:

Command Name	Code
Abort-Session	274
Accounting	271
Auth-Auth	265
Authentication-Info	318
Cancel-Location	317
Credit-Control	272
Delete-Subscriber-Data	320
Insert-Subscriber-Data	319
Notify	323
Purge-UE	321
Re-Auth	258
Reset	322
Session-Termination	275
Update-Location	316

Counts of these commands are available via SNMP and HDR output.

Other Diameter commands not listed above are counted in the Other category found in the aforementioned KPI displays.

Supporting Configuration

Supporting Configurations

The following configuration elements which are not mentioned in this guide are required for the Net-Net Diameter Director to function. Please refer to the Net-Net 4000 ACLI Configuration Guide for details about configuring all supporting elements.

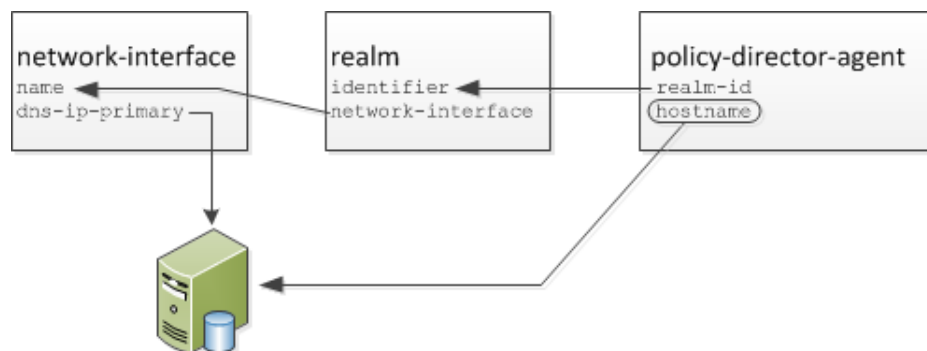
- network-interface
- phy-interface
- realm-config
- sip-config
- system-config

DNS for Diameter Director Agent Hostname Resolution

The Net-Net Diameter Director can use DNS to resolve FQDNs configured for each Diameter Director Agent. Configuring an FQDN for a Diameter Director Agent not only simplifies provisioning, but also eases network scalability and makes the network more resilient to node failure.

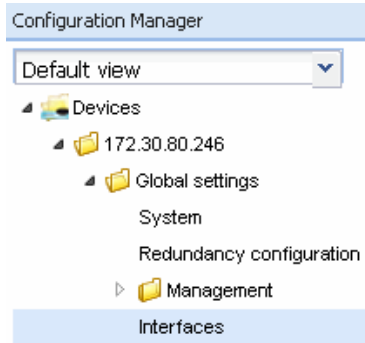
To enable the Net-Net Diameter Director to use FQDNs to resolve Diameter Director Agents, you must configure the network interface in which the Diameter Director Agent exists with DNS server information.

Ensure that the **realm**, which is configured in the *diameter director agent's realm id*, exists on the *network interface* as shown in the following diagram. The *diameter director agent's hostname* lookup will be sent to the *network interface's* DNS servers.



Configuring the Network Interface

1. Expand Configuration Manager > Global settings.
2. Click Interfaces.



3. Scroll to DNS.
4. **Primary**—Enter the primary DNS server's IP address. You can configure an additional two DNS servers by using the **First backup** and **Second backup** parameters.
5. **Default domain name**—Set the default domain name used to populate incomplete hostnames that do not include a domain for use with DNS queries. Entries must follow the Name format.
6. **DNS timeout**—Enter the total time in seconds you want to elapse before a query (and its retransmissions) sent to a DNS server would timeout. The default is 11 seconds.

Primary	172.30.80.100
First backup	
Second backup	
Default domain name	name
DNS timeout	11

7. Click **Apply**.

CPU Load Limiting

When experiencing a high CPU load, the Net-Net Diameter Director can throttle its response to incoming Diameter messages. This is set by configuring a CPU load value in the *diameter director config* configuration element's *load-limit* parameter.

CPU load limiting is a hard cut-off CPU threshold. Once the CPU exceeds a configured amount, incoming Diameter messages are rejected or dropped depending on how the *load limit action* parameter is configured as **drop** or **reject**.

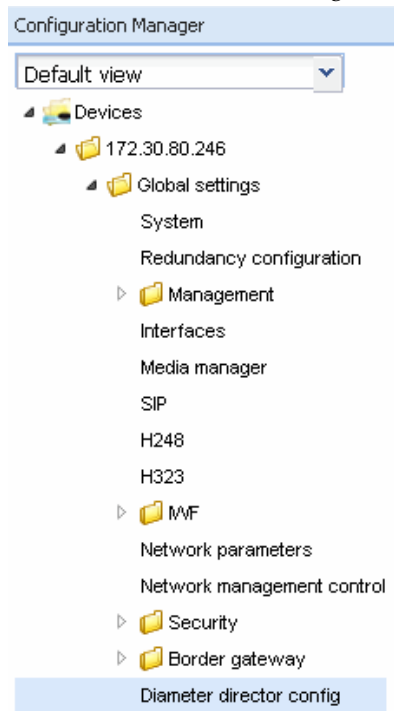
When the *load limit action* is set to **reject**, the Net-Net Diameter Director replies to the Diameter agent with an Answer response message including a valid error code. The error code is configured as a hexadecimal value in the *load limit result code* parameter. This value is inserted into the Result-Code AVP returned to the requesting agent.

DWR/DWA and CER/CEA messages are NOT subject to the CPU load limiting constraints. Ignoring a DWR/DWA or CER/CEA message could result in the connection being torn down, which might not be the desired action even in a high load situation. This is the Net-Net Diameter Director's action because maintaining the connection of peers is of extremely important.

Configuring Diameter Director Interfaces for Load Limiting

To configure a Diameter Director Interface for load limiting:

1. Expand Configuration Manager > Global settings.
2. Click Diameter director config.



3. **State**—Click **enabled** in the drop-down list to activate this CPU load limiting.
4. **Load limit**—Enter the CPU load value at which point load limiting is active.
5. **Load limit action**—Click either **reject** or **drop** in the drop-down list to perform on received Diameter messages once the *load limit* threshold has been crossed.

6. **Load limit result code**—Enter the value returned in the Result Code AVP returned to the requesting agent when the load limit has been exceeded and the action is set to **reject**. This should be set to a valid, decimal result code.

State	enabled ▼
Load limit	85
Load limit action	REJECT ▼
Load limit result code	3004

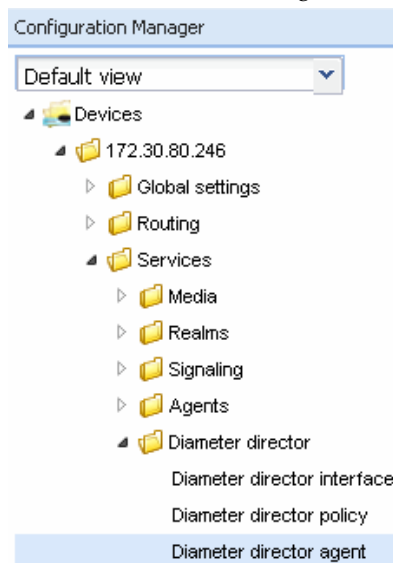
7. Click **Apply**.

Diameter Director On-Demand Peering

Configuring On-Demand Peering for Diameter Director

To configure on-demand peering:

1. Expand Configuration Manager > Services > Diameter Director.
2. Click Diameter Director agent.



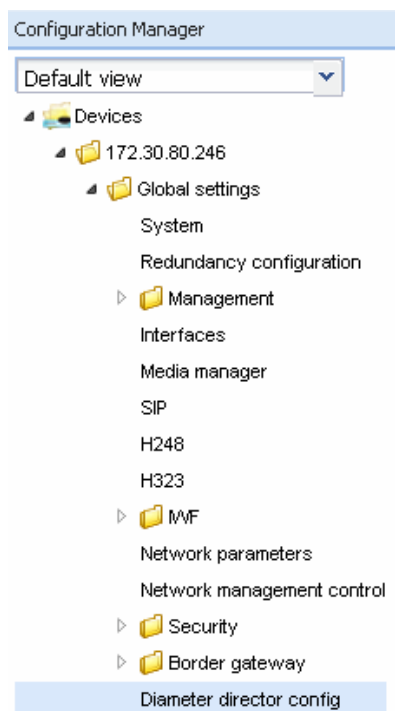
3. Click the Diameter Director agent you want to configure in the table and click **Edit**.
4. **Connection mode**—Click inbound in the drop-down list or keep the default value, outbound.
5. **On demand maximum inactivity timer**—This parameter is required for outbound-on-demand connection mode. Enter an interval in seconds. The valid range is **0-65535** seconds. The default value is **30** seconds.
6. Click **Apply**.

Session Statefulness

Configuring Stateful Policies

To configure stateful policies:

1. Expand Configuration Manager > Global settings.
2. Click Diameter Director config.



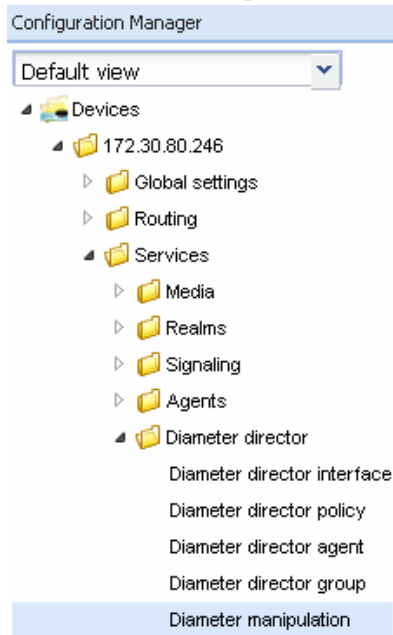
3. Press **Add** to add the configuration.
4. **Stateful policy**—Select the stateful policy for this Diameter director configuration.
5. Click **Apply**.

Diameter Director Generic Manipulation

Adding Manipulation Rules

To add Diameter Director generic manipulation rules:

1. Expand Configuration Manager > Services > Diameter Director.
2. Click Diameter manipulation.



3. Click the Diameter manipulation rule you want to configure in the table and click **Edit**.
4. Click **Add** at the Diameter manipulation rule table. The Add Diameter manipulation rule dialog box displays.
5. **Name**—Enter a unique name for this Diameter manipulation rule. This field is required.
6. **AVP code**—Enter an AVP code for this Diameter manipulation rule. The valid range is 0 -2147483647. The default value is 0.
7. Click **Apply**.

Configuring Diameter Manipulation Rules

To configure Diameter manipulation rules:

1. Click the Diameter manipulation rule you want to configure in the Diameter manipulation rule table and click **Edit**.
2. **Action**—Click an action in the drop-down list. This field is required. The valid action types are: replace, reject, store, delete, none, find-replace-all, add, log, and sip-manip. This field is required. The default value is **none**.
3. **Message command code**—Enter a message command code for this Diameter manipulation rule. The valid range is 0-2147483647. The default value is 0.
4. Click **Apply**.

LIR/LIA Lookup to Home Subscriber Server from I-BCF

The Net-Net Diameter Director, acting as an I-BCF, can perform a Location Information Lookup (LIR) to a Home Subscriber Server (HSS) to find the S-CSCF where to forward a request message. LIR/LIA messages are part of the Diameter Cx interface, which is used to communicate with HSS.

In an IMS environment, a user registers with S-CSCF and that S-CSCF assigns itself with an HSS as the serving registrar for that user. When a call needs to be routed to the user, an I-BCF performs a location information lookup to the HSS to determine the S-CSCF for that user and route the request toward that S-CSCF.

Home Subscriber Server

The Net-Net Diameter Director is configured with an HSS configuration element called home subscriber server. When the Net-Net Diameter Director receives an INVITE from a user, a local policy routing lookup is performed.

The home subscriber server configuration element is configured with a name, which is referenced in a local policy next-hop parameter. It also includes, a state, IP address, port, and realm.

P-Acme-Serving Parameter Creation

The Net-Net Diameter Director can insert a P-Acme-Serving parameter into the Route header of a message. This parameter is inserted when the parameter **Lookup parameter** is enabled in the home subscriber server configuration element and the Net-Net Diameter Director queried that HSS to find the message's destination.

The P-Acme-Serving parameter can be helpful for informing other Net-Net Diameter Directors (or Net-Net SMXs) that this Net-Net Diameter Director is acting as the UA's I-BCF and that the HSS lookup has been done.

HSS Watchdog Keepalives

The home subscriber server configuration element contains a **Watch dog timer** parameter that sets a time to send DWR messages to the HSS. This is similar to external policy server Diameter Heartbeat functionality. When 3/4 of this timer elapses, the Net-Net SBC begins disconnection processes. DWR message creation is suspended when any other activity with the HSS occurs within watchdog-ka-timer seconds. DWR creation resumes Watch dog timer seconds if all other traffic stops.

Local Policy

The Net-Net Diameter Director performs all expected processing on an incoming message and begins a local policy lookup. When it encounters a policy attribute with a next hop in the form of `cx:element-name`, it will perform an LIR/LIA transaction over the Cx Diameter interface to the home subscriber server referenced as element-name.

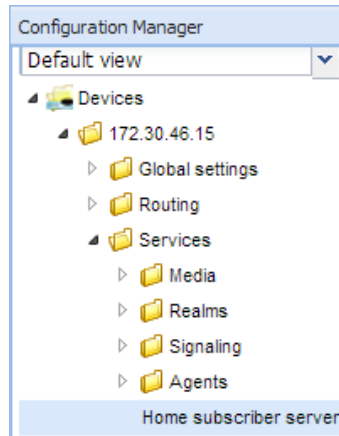
The Net-Net Diameter Director forwards the initial INVITE to the address in the Server-Name AVP returned by the HSS in the LIA message.

Configuration Instructions

Home subscriber server

To configure a home subscriber server (HSS):

1. Expand the Configuration Manager slider > Services.

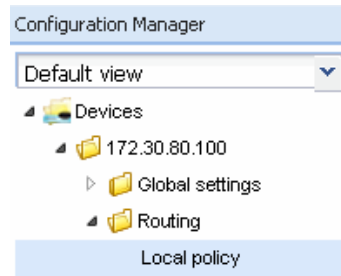


2. Click Home subscriber server. The Home subscriber server table opens in the content area.
3. Click **Add**.
4. **Name**—Enter the name for this home subscriber server configuration element to reference from other configuration elements.
5. **Address**—Enter the IP address of this HSS.
6. **Source realm**—Enter the realm name where this HSS exists.
7. Click **Apply**.
8. Select the Home Subscriber server from the table and click **Edit**.
9. **Port**—Enter the port number on the HSS where to connect. The default value is 80.
10. **Watch dog timer**—Enter the period of time to send DWRs to the HSS.
11. **Lookup parameter**—Set this to enabled for the Net-Net Diameter Director to add a P-Acme-Serving parameter to the Route header.
12. Click **Apply**.

Local Policy

To direct a policy attribute to retrieve a next hop from an HSS via an LIR lookup:

1. Expand the Configuration Manager slider > Routing.



2. Click Local policy. The local policy table appears in the content area.
3. Select an Local policy from the table and click **Edit**.
4. Scroll to the **Route policy attributes** table and click **Add**.
5. **Next hop**—Set this parameter to *cx:name-of-HSS-configuration* for this policy attribute to lookup routes by querying the named HSS via an LIR.
6. Click **Apply**.

Acronym List

General Use Acronyms

3GPP— 3rd-Generation Partnership Project

A

AAA—Administration, Authentication, and Authorization

ACD—Automatic Call Distribution

ACL—Access Control List

ACLI—Acme Command Line Interface

ACP—Acme Control Protocol

ACK—Acknowledgement (SIP)

ADMF—ADMinistration Function

ADSL—Asymmetric Digital Subscriber Line

AF—Access Function

AFID—Access Function Identifier

AIN—Advanced Intelligent Network

ALG—Application Layer Gateway

ANI —Automatic Number Identification (ISDN)

ANSI—American National Standards Institution

AoR—Address of Record

AP—Application Protocol

API—Application Programming Interface

ANP—Access Point Name

APPN—Advanced Peer-to-Peer Networking

ARP—Address Resolution Protocol

ARQ—Admission Request (H.323)

ASCII—American Standard Code for Information Interchange

ASIC—Application-Specific Integrated Circuit

ASN—Abstract Syntax Notation

ASN.1—Abstract Syntax Notation – 1

ASP—Application Service Provider, Active Server Pages, Adjunct Service Point

ASR —Access Service Request

ATCP—Async TCP

ATCP stack—Refers to the separate TCP stack implemented in the ATCP task.
 ATCP task—Refers to the task in the system in which the ATCP stack executes.
 ATCP socket—Refers to a socket used for an async TCP connection.
 ATM—Asynchronous Transfer Mode
 ATX—Advanced Technology Extended

B

B2BUA—Back-to-Back User Agent
 BBSRAM—Battery Backup Static Random Access Memory
 BC—Bearer Capability
 BCID—Billing Correlation Identifier
 BER—Basic Encoding Rules
 BGF—Border Gateway Function
 BGP—Border Gateway Protocol
 BHCA—Busy Hour Call Attempts
 BIOS— Basic Input/Output System
 BIS—Bearer-Independent Setup
 BITS—Building Integrated Timing Supply
 B-ICI—Broadband Inter-carrier Interface (ATM)
 B-ISUP—Broadband ISDN User Part
 BNF—(augmented) Backus-Naur Form
 BoS—Bottom of Stack
 bps—Bits per Second
 BRAS—Broadband Remote Access Server
 BRI—Basic Rate Interface (ISDN)
 BSP— Board Support Package
 BTU—British Thermal Units

C

CA—Certificate Authority
 CAC—Connection Administration Control (ATM)
 CALEA—Communications Assistance to Law Enforcement Agencies
 CAM—Content Addressable Memory
 CARP—Cache Array Routing Protocol (to replace ICP)
 CAS—Cordless Access Service
 C-BGF—Core Border Gateway Function
 CBR—Constant Bit Rate (ATM)
 CC —Country Code/ Call Content

CCC—Call Content Connection/ Handover Interface 3 (Call Content)
CCCI—Call Content Connection Identifier
CCM—Cisco Call Manager
CD—Call Data
CDC—Call Data Connection/ Handover Interface 2 (Call Data)
CF—Collection Function
CDPN—Called Party Number
CDR—Call Detail Record
CE—Conformité Européenne (The CE marking is a European proof of conformity and is also described as "passport" that allows manufacturers and exporters to circulate products freely within the EU.)
CFM—Cubic Feet per Minute (fan speed)
CFU—Call Forwarding Unconditional
CGI—Common Gateway Interface
CIC—Carrier Indicator Code/Carrier Identification Code
CID—Caller Identification
CISSP—Certified Information Systems Security Professional
CLC—Close Logical Channel
CLCAck—Close Logical Channel Ack
Client-SI—Client-Server Information
CLEC—Competitive Local Exchange Carrier
CLI—Command-line Interface
CMIP—Common Management Information Protocol
CMIS—Common Management Information Service
CMOT—Common Management Information Services Over TCP
CMS—Call Management Server
CNM—Customer Network Management
CO—Connection Oriented
CODEC—Coder/Decoder
COPS—Common Open Policy Service
CORBA—Common Object Request Broker Architecture
CoS—Class of Service
CP—Communications Processor
CPL—Call Processing Language
CPLD—Complex Programmable Logic Device
CPM—Communications Processor Module
CPU—Central Processing Unit

CRA—Call Routing Apparatus
 CRI—Call Related Information
 CRLF—Carriage Return Line Feed
 CS—Circuit Switch
 CSA—Client Server Architecture
 CSPDN—Circuit Switched Public Data Network
 CSU—Channel Service Unit
 CT—Cordless Telephone
 CT-1—European analogue cordless telephone system
 CT-2—Second-generation cordless telephone, Digital
 CTI—Computer Telephony Integration
 CUG—Closed User Group

D

DA—Destination Address
 DAM—Data Access Method; Data Asset Management
 DDD—Direct Distance Dialing
 DDF—Digital Distribution Frame
 DECT—Digital European Cordless Telephone
 DER—Distinguished Encoding Rules
 DES—Data Encryption Standard
 DF—Distribution Function
 DHCP—Dynamic Host Configuration Protocol
 DIAMETER—a protocol for authentication, authorization, and accounting
 DiffServ—Differentiated Services
 DIMM—Dual In-line Memory Module
 DLCI—Data Link Connection Identifier
 DLSR—Delay Since Last Send Report
 DN—Directory Number
 DNS—Domain Name Server/Service
 DOM—Document Object Model
 DoS—Denial of Service
 DP—Destination Port
 DPCM—Differential Pulse Code Modulation
 DRAM—Dynamic Random Access Memory
 DS—Differentiated Services
 DSA—Digital Signature Algorithm

DSCP—DiffServ Codepoint
 DSL—Digital Subscriber Line
 DSLAM—Digital Subscriber Line Access Multiplexer
 DSP—Digital Signal Processing
 DSS—Digital Satellite System
 DSU—Digital Service Unit
 dTCP—Dynamic Transmission Control Protocol
 DTD—Document Type Definition
 dTLS—Dynamic Transport Layer Security
 DTMF—Dial Tone Multi-Frequency

E

ED—Ending Delimiter
 EEPROM—Electrically Erasable Programmable Read-Only Memory
 EFTPOS—Electronics Funds Transfer Point of Sale
 EGP—Exterior Gateway Protocol
 EMB—Early Media Blocking
 EMC—Electromagnetic Compatibility
 EMI —Electromagnetic Interference
 ENUM—Refers to the use of an E.164 number, in reverse, with domain notation (i.e., dotted).
 EPROM—Erasable Programmable Read-Only Memory
 ER—Edge Router
 ESD—Electrostatic Discharge
 ETSI—European Telecommunications Standards Institute

F

FCC—Federal Communications Commission
 FCP—Firewall Control Protocol
 FEC—Forward Equivalence Class
 FPGA—Field Programmable Gate Array
 FQDN—Fully Qualified Domain Name
 FS—Fast-start
 FSA—Foreign SIP Agent (ACME-specific term?)
 FTP—File Transfer Protocol
 FTR—Flow Transform Record

G

GA—Global Address

GB—Gigabyte
 GBPS—Gigabits Per Second
 GigE—Gigabit Ethernet
 GMII—Gig Media Independent Interface
 GNU—GNUs not UNIX
 GOSIP—Government Open Systems Interconnection Profile
 GPS—Global Policy Server/Global Positioning System
 GSA—Global System Administrator
 GSM—Global Systems for Mobile Communications
 GSTN—Global Switched Telephone Network

H

HA—High Availability (Acme Packet redundancy solution)
 HNT—Hosted NAT Traversal (Acme Packet)
 HTML—Hypertext Markup Language
 HTTP—Hypertext Transport Protocol

I

I/O—Input/Output
 IAD—Integrated Access Device
 IANA—Internet Assigned Numbers Authority
 IAP—Intercept Access Point
 I-BCF—Interconnect Border Control Function
 I-CSCF—Interrogating Call Session Control Function
 ICE—In Circuit Emulator
 ICMP—Internet Control Message Protocol
 ICP—Internet Cache Protocol
 IDS—Intrusion Detection System
 IE—Information Element
 IEC—International Electrotechnical Commission
 IEEE—Institute of Electrical and Electronics Engineers
 IESG—Internet Engineering Steering Group
 IETF—Internet Engineering Task Force
 IGP—Interior Gateway Protocol
 IIS—Internet Information Server
 ILEC —Independent Local Exchange carrier (USA)
 IMS—IP Multimedia Subsystem
 IN—Intelligent Network

IOS—Internetworking Operating System
 IP—Internet Protocol (IPv4, IPv6)
 IPC—Inter-process Communication
 IPDR—Internet Protocol Data Record
 IPSec—Internet Protocol Security
 IPtel—Internet Protocol Telephony
 IPv—Internet Protocol version
 IS—Intercept Server
 ISDN—Integrated Services Digital Network
 ISO—International Organization of Standardization
 ISP—Internet Service Provider
 ITAD—Internet Telephony Administrative Domain
 ITSP—Internet Telephony Service Provider
 ITU—International Telecommunication Union
 ITU-T—International Telecommunication Union Telecommunication
 Standardization
 IVR—Interactive Voice Response
 IWF—Interworking Function (referring to the Net-Net 4000's SIP-H.323
 interworking)
 IXC—Interexchange Carrier

J

JTAG—Joint Test Action Group
 JTAPI—Java telephony application programming interface

K

Kb—Kilobits
 KB—Kilobytes
 Kbps—Kilobits per second
 KEA—Key Exchange Algorithm
 KTS—Key Telephone System

L

LA—Local Address
 LAES—Lawfully Authorized Electronic Surveillance
 LAN—Local Area Network
 LATA—Local Access and Transport Area
 LCD—Liquid Crystal Display
 LDAP—Lightweight Direct(ory) Access Protocol

LEA—Law Enforcement Agency
 LEAF—Law Enforcement Administrative Function
 LEC—Local Exchange Carrier
 LED—Light Emitting Diode
 LEM—Local Element Manager
 LEN—Local Exchange Node
 LI—Lawful Intercept/ Legal Interception
 LNP—Local Number Portability
 LOS—Loss of Signal
 LS—Location Server
 LSB—Least Significant Bit
 LSR—Label-switching router

M

MAC— Media Access Control/ Message Authentication Code
 MAN— Metropolitan Area Networks
 Mb—Megabits
 MB—Megabytes
 MBCD—Middlebox Control Daemon
 Mbone—Multicast Backbone
 MC—Monitoring Center
 MCU—Multi-party Conference Unit
 MD5—Message Digest 5 (hash function)
 MF—Media Firewall/ Mediation Function
 MG—Media Gateway
 MGCP—Media Gateway Communication Protocol
 MHz—Megahertz
 MIB—Management Information Base (TMN)
 MIB II—Management Information Base II
 MIBOC —Middlebox Control Protocol
 MIDCOM—Middle Box Communications
 MIME—Multipurpose Internet Mail Extension
 MOC—Mandatory, Optional, Conditional
 MoIP—Messaging over Internet Protocol
 MP—Main Processor
 μ P—Microprocessor (subsystem)
 MPLS—Multi-protocol Label Switching

MR—Media Router
 MRCP—Media Router Control Protocol
 MSB—Most Significant Bit
 MSD—Master-Slave Determination
 MTA—Message Transfer Agent
 MTA—Multimedia Terminal Adapter
 MTBF—Mean Time Between Failures
 MTTR—Mean Time To Repair
 MTU—Maximum Transmission Unit
 MX—Mail Exchange

N

N-ACD—Network Automotive Call Distribution
 NANP—North American Numbering Plan
 NAPT—Network Address Port Translation
 NAS—Network Access Security
 NAT—Network Address Translation
 Nco—Network Code of Practice
 NCP—Network Control Point
 NCS—Network Call Signaling
 NEBS—Network Equipment - Building Systems/Standards
 NE—Network Element
 NIC—Network Interface Card
 NMS—Network Management Station
 NNC—Net-Net Central
 NP—Network Processor
 NSRG—Network Signaling Record Generator
 NTE —Networking Terminating Equipment
 NTP—Network Time Protocol
 NTU—Networking Terminating Unit
 NVRAM—Non-volatile Random Access Memory

O

OAM—Operation, Administration, and Maintenance
 OC—Optical Carrier
 OC-n—Optical Carrier transport
 OCx—Optical Carrier level
 OEI—Optical Electrical Interface

OEM—Original Equipment Manufacturer
 OID—Object Identifier
 OLC—Open Logical Channel
 OLCAck—Open Logical Channel Ack
 ONP—Open Network Provision
 OS—Operating System
 OSI—Open System Interconnect(ion)
 OSP—Open Settlement Protocol
 OSPF—Open Shortest Path First
 OSS—Operation Support Systems

P

PABX—Private Automatic Branch Exchange
 PACS—Personal Access Communications Systems
 PAT—Port Address Translation
 PBX—Private Branch Exchange
 P-CSCF—Proxy Call Session Control Function
 PCB—Printed Circuit Board
 PCDATA—Parseable Data Characters
 PCI—Peripheral Component Interconnect
 PCMCIA—Personal Computer Memory Card International Association
 PCN—Personal Communications Network
 PCS—Personal Communications Services
 PD—Packet Data
 PDCS—Packet Cable Distributed Call Signaling
 PDH—Plesiochronous Digital Hierarchy
 PDN—Public Data Network
 PDN—Packet Data Network
 PDP—Policy Decision Point
 PDU—Protocol Data Unit (or Packet Data Unit)
 PEM—Privacy Enhanced Mail
 PEP—Policy Enforcement Point/Protocol Extensions Protocol
 Perl—Practical Extraction Report Language
 PHY—Physical Layer Device
 PIB—Policy Information Base
 PING—Packet Internet Groper
 PINT—PSTN and IP Internetworking

PKCS-7—RFC 2315, Cryptographic Message Syntax, Version 1.5
 PKCS-10—RFC 2314, Certificate Request Syntax, Version 1.5
 PKI —Public Key Infrastructure
 PMC—PCI Mezzanine Card
 PNNI—Private Network Node Interface (ATM)
 PNO—Public Network Operator
 POP—Point of Presence
 POS—Packet Over SONET
 POTS—Plain Old Telephone Service
 PPP—Point-to-Point Protocol
 PROM—Programmable Read-Only Memory
 PSTN—Public Switched Telephone Network (Telecom Network)
 PTE—Packet Transform Engine
 PTO—Public Telecommunications Operator
 PTT—Post, Telephone, and Telegraph
 PWB—Printed Wiring Board

Q

QoS—Quality of Service
 QSIG—Unified International Digital Corporate Network Signaling Standard

R

RADIUS—Remote Authentication Dial-in User Service
 RAM—Random Access Memory
 RARP—Reverse Address Resolution Protocol
 RAS—Remote Access Service; Registration Admission and Status (H.323)
 RC2 and RC4—Rivest encryption ciphers developed for RSA Data Security
 RED—Random Early Discard
 REN—Ringer Equivalent Number
 RFC—Request for Comments
 RIP—Routing Information Protocol
 RISC —Reduced Instruction Set Chip
 RMON—Remote (Network) Monitoring
 ROM—Read-Only Memory
 RPC—Remote Procedure Call
 RR—Received Report
 RS-232—Recommended Standard 232 (computer serial interface, IEEE)
 RSA—Rivest, Shamir, & Adleman (public key encryption technology)

RSIP—ReStart In Progress
 RSVP—Resource Reservation Protocol
 RTCP—Real-time Control Protocol
 RTP—Real-time Transport Protocol
 RTP/AVP—Real-time Transport Protocol/Audio-Video Protocol
 RTSP—Real-time Streaming Protocol
 RTT—Round Trip Time

S

SA—Source Address
 SA—Session Agent
 SAG—Session Agent Group
 SBC—Session Border Controller
 S-CSCF—Serving Call Session Control Function
 SCE—Service Control Environment
 SCP—Service Control Point
 SCTP—Streaming Control Transmission Protocol
 SD—(Net-Net) Session Director
 SDH—Synchronous Digital Hierarchy
 SDP—Session Description Protocol
 SDRAM—Synchronous Dynamic Random Access Memory
 SERDES—Serial De-serializer
 SHA-1—Secure Hash Algorithm, a hash function used by the U.S. Government
 SIG—Special Interest Group
 SIM—Subscriber Identity Module
 SIMM—Single In-line Memory Module
 SIP—Session Initiation Protocol
 SLA—Service Level Agreement
 SME—Small to Medium Enterprise(s)
 SMIL—Synchronized Multimedia Integration Language
 SMP—Simple Management Protocol
 SMTP—Simple Mail Transfer Protocol
 SNMP—Simple Network Management Protocol
 SOCKS—SOCKeT server
 SONET—Synchronous Optical Network
 SP—Source Port / Service Provider
 SPAF—Service Provider Administrative Function

SR—(Net-Net) Session Router
 SRAM—Static Random Access Memory
 SRS—Session Routing System
 SRV—Resource record for servers (DNS)
 SS—Slow-start
 SS7—Signaling System 7
 SSH—Secured Shell or Secure Socket Shell
 SSL—Secure Socket Layer
 SSP—Service Switching Point
 sTCP—Static Transmission Control Protocol
 STL—Standard Template Library
 sTLS—Static Transport Layer Security
 STP—Signal Transfer Point; Service Transfer Point
 SVC—Signaling Virtual Channel (ATM)
 SVC—Switched Virtual Circuit (Packet Switching)

T

TA—Terminal Adapter (ISDN)
 TAC—Terminal Access Control
 TACACS+—Terminal Access Controller Access Control System
 TAPI—Telephony Application Program Interface
 TAXI—Transparent Asynchronous Transmitter/Receiver Interface
 TCB—Task Control Bar/Task Control Block
 TCI—Tag Control Identifier
 TCP—Transmission Control Protocol
 TCP/IP—Transmission Control Protocol /Internet Protocol
 TCS—Terminal Capability Set
 TEN—Transit Exchange Node
 TFTP—Trivial File Transfer Protocol
 TLS—Transport Layer Security (same as SSL)
 TLV —Tag Length Value
 TM—Traffic Manager
 TMN—Telecommunications Management Network
 ToS—Type of Service
 TRIB—Telephony Routing Information Base
 TRIP—Telephony Routing over IP
 TS—Time Slot

TSAP—Transport Service Access Point
TSAPI —Telephony Server API
TTL—Time to Live

U

UA—User Agent
UAC—User Agent Client
UAS—User Agent Server
UDP—User Datagram Protocol
UE—User Equipment
UL—Underwriters Laboratories
UMTS—Universal Mobile Telecommunications Systems
UNI—User-to-Network Interface
UPS—Uninterruptible Power Supply
UPT—Universal Portable Telephone
URI—User Resource Identifier
URL—Uniform Resource Location
UTC—Coordinated Universal Time
UTP—Unshielded Twisted Pair

V

VAC—Volts Alternating Current
VANS—Value Added Network Services
VAR—Value Added Reseller
VarBind—Variable Binding
VBR—Variable Bit Rate
VC—Virtual Channel (ATM)/Virtual Container (SDH)
VCC—Virtual Channel Connection (ATM)
VCI—Virtual Channel Identifier
VDC—Volts Direct Current
VFD—Vacuum Florescent Display
VLAN—Virtual Local Area Network
VLL—Virtual Leased Lines
VoIP—Voice Over Internet Protocol
VP—Virtual Path
VPC—Virtual Path Connection
VPI—Virtual Path Identifier
VPN—Virtual Private Network

VSA—Vendor-specific Attribute (RADIUS extension)

VTOA—Voice and Telephony over ATM

W

WAN—Wide Area Network

WLL—Wavelength Division Multiplex

X

XE—Translation Engine

XML—Extensible Markup Language

XSM—External Search Machine

Y

(None to list.)

Z

((None to list.)

Signaling Protocol Acronyms

The acronyms used in this guide's discussion of H.323 signaling services and IWF services.

H.323

We use the acronyms listed below to refer to H.323 signaling messages and other related H.323 behavior.

ACF—Admission Confirm

Alerting—Message used when called party alerted

ARQ—Admission Request

Call Proceeding—Message used when call established

CLC—Close Logical Channel

CLC Ack—Close Logical Channel Acknowledgment

Connect—Message used when called party accepts call

GRQ—Gatekeeper Discovery

IRQ—Information Request

IRR—Information Request Response

LCF—Location Confirm

LRJ—Location Reject

LRQ—Location Request

M/S Det.—Master/Slave Determination

OLC—Open Logical Channel

OLC Ack—Open Logical Channel Acknowledgment

RAS—Registration, Admission, and Status
RCF—Registration Confirm
Release Complete—Message used when call is released, signaling channel open
RRJ—Registration Reject
RRQ—Registration Request
Setup—Message used to request connection
TCS—Terminal Capability Set
UCF—Unregistration Confirm
URJ—Unregistration Reject
URQ—Unregistration Request