

**Oracle® Communications Core Session
Manager**

Essentials Guide

Release S-CZ6.3.15M1

October 2013

Copyright ©2013, 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	ix
Overview	ix
Audience	ix
About Net-Net Software Releases	ix
Related Documentation	ix
Release Features	x
Release Caveats	x
Revision History	xii
1 Diameter Based Oracle CSM	12
Message Authentication for SIP Requests	12
User Authorization	12
SIP Digest User Authentication	13
SIP Authentication Challenge	13
SIP Authentication Response	14
Oracle CSM Authentication Check.....	14
Oracle CSM as Registrar	15
New Registration	15
Limiting AOR Contacts	15
HSS Server Assignment	16
Server Assignment Messages	16
Register Refresh	16
Entry Unregistration	17
User Registration based on Reg-ID and Instance-ID (RFC 5626)	18
Outbound Registration Binding Processing.....	19
Wildcarded PUID Support	19
ACLI Instructions	20
SIP Registration Event Package Support	23

SUBSCRIBE Processing	24
SUBSCRIBE REFRESH Requests	25
Reg Event NOTIFY Messages	25
Reducing NOTIFY Traffic	26
Configuring Registration Event Package	27
ACLI Instructions	27
Message Routing	28
Registrar Routing	28
Default Egress Realm	28
Routing Based on UA Capabilities	29
ACLI Instructions	31
Tel-URI Resolution	32
HSS Initiated User Profile Changes	34
Other Diameter Cx Configuration	34
Host and Realm AVP Configuration for Cx	34
ACLI Instructions	35
Initial Filter Criteria (IFC)	35
SIP Registration	36
SIP Call	36
Evaluating Session Case in the P-Served-User Header	37
Supported Sessioncase and Registration State	37
Additional Options	38
IFC Support for Unregistered Users	38
UE-terminating requests to an unregistered user	38
Caching the Downloaded IFC	39
Optimizing IFC Updates	39
Push Profile Request (PPR) updates	40
ACLI Instructions	40
Shared and Default iFCs	41
SiFC Usage	41
DiFC Usage	41
SiFC/DiFC File Example	42
iFC Execution Order	43
Refreshing SiFC and DiFC Files	43
ACLI Instructions	43
Distinct and Wildcarded Public Service Identity (PSI) Support	44
Configuring SIP Ping OPTIONS Support	44
ACLI Instructions	44
Redundancy and Load Balancing with HSS Servers	45

About HSS Groups	45
Connection Failure Detection	46
2 ENUM Based Oracle CSM	48
Message Authentication for SIP Requests	48
Credential Retrieval	48
User Authentication Query	49
SIP Digest User Authentication	49
SIP Authentication Challenge	49
Authentication Header Elements	49
SIP Authentication Response	49
Oracle CSM Authentication Check	49
Oracle CSM as Registrar	50
DDNS Update to User Subscriber Database	50
ENUM Database Correlation	51
Register Refresh	52
Limiting AOR Contacts	53
User Registration based on Reg-ID and Instance-ID (RFC 5626)	53
Outbound Registration Binding Processing	54
ENUM Database Update	54
NAPTR Update Format	54
ACLI Instructions	55
OAuth 2.0 Support	58
OAuth Operation	58
Configuring OAuth Support	59
ACLI Instructions	60
Enabling the SPL Plug-in	60
Configuring the Plug-in Option	62
Message Routing	62
Registrar Routing	62
Default Egress Realm	63
ACLI Instructions	63
Tel-URI Resolution	63
Extended ENUM Record Length	65
NAPTR and TXT Record Creation and Association	66
NAPTR Record Format	66
TXT Record Retrieval	66
Requirements	67
SIP User Parts - RFC 3261 Character Set Support	67

Encoding Alpha-Numerics	67
Multiple DNS Zone Support	67
ACLI Instructions	68
Configuring SIP Ping OPTIONS Support	68
ACLI Instructions	68
3 Local Subscriber Tables	70
Local Subscriber Table	70
LST Runtime Execution	70
LST Configuration	70
ACLI Instructions	71
LST Redundancy for HA Systems	71
Reloading the LST	72
LST File Compression	72
LST File Format	72
LST Subscriber Hash and Encryption	73
4 Third Party Registration	76
Third Party Registrations via iFCs	76
Embedded REGISTER	77
ACLI Instructions - Third Party Registration via iFCs	77
Session Agent	77
SIP Registrar	78
Third Party Registration via ACLI Configuration	79
Third Party Registration Server States	79
Third Party Registration Expiration	80
Defining Third Party Servers	80
ACLI Instructions - Third Party Server Configuration	80
Third Party Registrar	80
SIP Registrar	81
5 RADIUS Accounting of REGISTERS	84
CDR Generation for REGISTER Events	84
REGISTER Scenarios	84
REGISTER VSA Format	87
ACLI Instructions	88
Example CDRs	88
Local CDR CSV Orientation	92

6	Reference and Debugging	108
	ACLI Configuration Elements	108
	sip-registrar	108
	Parameters	108
	Path	109
	sip-authentication-profile	109
	Parameters	109
	Path	110
	home-subscriber-server	110
	Parameters	110
	Path	111
	third-party-regs	111
	Parameters	111
	Path	111
	local-subscriber-table	111
	Parameters	111
	Path	112
	enum-config	112
	Parameters	112
	Path	113
	ifc-profile	113
	Parameters	113
	Path	113
	regevent-notification-profile	113
	Parameters	113
	Path	114
	SNMP MIBs and Traps	114
	Acme Packet License MIB (ap-license.mib)	115
	Acme Packet System Management MIB (ap-smgmt.mib)	115
	Enterprise Traps	115
	Oracle CSM Show Commands	115
	Supporting Configuration	124
	Verify Config	124
	sip authentication profile (CX)	124
	sip authentication profile (ENUM)	125
	sip authentication profile (Local)	125
	sip-registrar	125
	sip-registrar	125

Memory Utilization125

About This Guide

Overview

Oracle® Communications Core Session Manager (CSM) applies core session control to reduce the complexity and cost of delivering high-value, revenue generating SIP multimedia services. Oracle CSM can be used to support a broad range of SIP services including residential or business voice, GSMA-defined Rich Communication Suite (RCS) services and fixed mobile convergence (FMC) for small subscriber populations or initial service rollouts.

Audience

This guide is written for network administrators and those who configure network devices. This document is meant to be used in conjunction with the Net-Net 4000 CLI Configuration guide in order to provide full explanation of all functionality. Strong familiarity with the Oracle Communications SBC is required for use with this guide.

The Oracle Core Sesion Manager Essentials Guide provides information related to the features, operation, and maintenance of the Oracle Core Sesion Manager Essentials Guide. Only experienced and authorized personnel should perform installation, configuration, and maintenance tasks.

About Net-Net Software Releases

Release Version S-Cz6.3.15M1 is supplied as virtual machine software or as a software-only delivery suitable for operation on server hardware. Refer to sales documentation updates for information further specifying hardware support.

Release Version S-Cz6.3.15M1 is the first release of the Oracle CSM product.

Note that, in contrast to generic virtual machine installation documentation, the minimum required memory for the Oracle CSM is 4G.

Related Documentation

The following table lists the members that comprise the documentation set for this release:

Document Name	Document Description
Acme Packet 4500 System Hardware Installation Guide	Contains information about the components and installation of the Acme Packet 4500 system.
Acme Packet 3800 Hardware Installation Guide	Contains information about the components and installation of the Acme Packet 3800 system.
Release Notes	Contains information about the current documentation set release, including new features and management changes.
ACLI Configuration Guide	Contains information about the administration and software configuration SBC.

Document Name	Document Description
ACLI Reference Guide	Contains explanations of how to use the ACLI, as an alphabetical listings and descriptions of all ACLI commands and configuration parameters.
Maintenance and Troubleshooting Guide	Contains information about SBC logs, performance announcements, system management, inventory management, upgrades, working with configurations, and managing backups and archives.
MIB Reference Guide	Contains information about Management Information Base (MIBs), Enterprise MIBs, general trap information, including specific details about standard traps and enterprise traps, Simple Network Management Protocol (SNMP) GET query information (including standard and enterprise SNMP GET query names, object identifier names and numbers, and descriptions), examples of scalar and table objects.
Accounting Guide	Contains information about the SBC's accounting support, including details about RADIUS accounting.
HDR Resource Guide	Contains information about the SBC's Historical Data Recording (HDR) feature. This guide includes HDR configuration and system-wide statistical information.
Administrative Security Essentials	Contains information about the SBC's support for its Administrative Security license.

Release Features

This section provides a table listing content in Net-Net OS Release Version S-Cz6.3.15M1.

Feature	Description
Registration Information distribution for third party registrations	Feature to iFC related third party registrations that supplies registration information in MIME of applicable messages when required.

Release Caveats

The following sections list caveats related to Version S-Cz6.3.15M1 of the Oracle CSM.

- The ISC interface does not work when dialog transparency is enabled on the Oracle CSM
 - Resolution - Do not enable dialog transparency if your Oracle CSM must support ISC.

- The Oracle CSM does not work with an iFC when its default handling is set to “SESSION CONTINUED”.
- When system memory usage on the Oracle CSM exceeds 50%, the command `show cpu top` fails, producing the error message “ERROR: could not fork”.
 - Issue - The `show cpu top` command does not work during high memory utilization.
 - Resolution - To see process resource utilization during periods when the system is using more than 50% memory, use the `show processes cpu` command.
- Multi-stage routing does not work for S-CSCF routing functions.
- The Oracle CSM accepts only the first message received from an application server in response to messages from the Oracle CSM that included an ODI. The Oracle CSM drops any subsequent messages with the same ODI.
 - Resolution - Do not configure an AS to fork responses to the Oracle CSM that include an ODI originally provided by the Oracle CSM.
- The Oracle CSM is available for operation over OVM with limited functionality.
- A Oracle CSM registrar set to DDNS that also has the following option set does not work properly:
`e164-<primary|secondary>-config=enum:<enum-config-name>`
 - Issue - Rather than forwarding a call via the DDNS database, the Oracle CSM is referring to local policy configuration and forwarding the call to the next hop that results from the local policy lookup.
 - Resolution - Do not configure the option shown above on a registrar that operates with a DDNS database (`subscriber-database-method` set to DDNS).
- The Oracle CSM requires 4G memory. Ensure you have adequate memory before installing Net-Net CSM software.
 - Issue - Generic virtual machine installation documentation specifies the minimum memory for Net-Net software as 2G.
 - Resolution - Install the Oracle CSM on virtual machines with a minimum of 4G memory.
- By default, storage is not persistent across a reboot of a Oracle CSM virtual machine. You must create persistent storage space for log and dump file data.
 - Issue - Generic virtual machine installation documentation may not include the requirement to run the command `format hard-disk` during virtual machine installation.
 - Resolution - Run the command `format hard-disk` to create a persistent partition for your `/opt` directory, within which you can store data needed after a reboot. Perform this procedure the FIRST time you start your Oracle CSM.

- The Oracle CSM does not send third party registration for the entire implicit registration set. It only sends this for the specific public user identity that is registering, de-registering or re-registering.

Revision History

This section contains a revision history for this document.

Date	Revision Number	Description
Oct 4, 2013	1.00	<ul style="list-style-type: none">• Initial Release
Oct 17, 2013	1.10	<ul style="list-style-type: none">• Removes caveat related to the device using iFC priority to determine trigger order.• Adds caveat specifying the device send third party registrations for specific PUIs, not for an entire implicit registration set.

The Oracle CSM functions in an IMS core. It communicates with the HSS to obtain Authorization, Authentication, S-CSCF assignment, and ultimately routing instructions. To accomplish these functions, the Oracle CSM can perform the SIP registrar role in conjunction with an HSS.

Message Authentication for SIP Requests

The Oracle CSM authenticates requests by configuring the *sip authentication profile* configuration element. The *name* of this configuration element is either configured as a parameter in the *sip registrar* configuration element's *authentication profile* parameter or in the *sip interface* configuration element's *sip-authentication-profile* parameter. This means that the Oracle CSM can perform SIP digest authentication either globally, per domain of the Request URI or as received on a SIP interface.

After naming a *sip authentication profile*, the received methods that trigger digest authentication are configured in the *methods* parameter. You can also define which anonymous endpoints are subject to authentication based on the request method they send to the Oracle CSM by configuring in the *anonymous-methods* parameter. Consider the following three scenarios:

1. By configuring the *methods* parameter with REGISTER and leaving the *anonymous-methods* parameter blank, the Oracle CSM authenticates only REGISTER request messages, all other requests are unauthenticated.
2. By configuring the *methods* parameter with REGISTER and INVITE, and leaving the *anonymous-methods* parameter blank, the Oracle CSM authenticates all REGISTER and INVITE request messages from both registered and anonymous endpoints, all other requests are unauthenticated.
3. By configuring the *methods* parameter with REGISTER and configuring the *anonymous-methods* parameter with INVITE, the Oracle CSM authenticates REGISTER request messages from all endpoints, while INVITES are only authenticated from anonymous endpoints.

User Authorization

In an IMS network, the Oracle CSM requests user authorization from an HSS when receiving a REGISTER message. An HSS is defined on the Oracle CSM by creating a *home subscriber server* configuration element that includes a *name*, *ip address*, *port*, and *realm* as its basic defining data.

UAR/UAA Transaction

Before requesting authentication information, the Oracle CSM sends a User Authorization Request (UAR) to the HSS for the registering endpoint to determine if this user is allowed to receive service. The Oracle CSM populates the UAR's AVPs as follows:

- Public-User-Identity—the SIP AOR of the registering endpoint
- Visited-Network-Identity—the value of the *network-id* parameter from the ingress *sip-interface*.

- Private-User-Identity—the username from the SIP authorization header, if it is present. If not, this value is the public User ID.
- User-Authorization-Type—always set to REGISTRATION_AND_CAPABILITIES (2)

The Oracle CSM expects the UAA to be either:

- DIAMETER_FIRST_REGISTRATION
- DIAMETER_SUBSEQUENT_REGISTRATION

Any of these responses result in the continued processing of the registering endpoint. Any other result code results in an error and a 403 returned to the registering UA (often referred to as a UE). The next step is the authentication and request for the H(A1) hash.

SIP Digest User Authentication

Authentication via MAR/MAA

To authenticate the registering user, the Oracle CSM needs a digest realm, QoP, and the H(A1) hash. It requests these from a server, usually the HSS, by sending it a Multimedia Auth Request (MAR) message. The MAR's AVPs are populated with:

- Public-User-Identity—the SIP AOR of the endpoint being registered (same as UAR)
- Private-User-Identity—the username from the SIP authorization header or the SIP AOR if the *AOR for PUID* parameter is enabled. (Same as UAR)
- SIP-Number-Auth-Items—always set to 1
- SIP-Auth-Data-Item -> SIP-Item-Number—always set to 1
- SIP-Auth-Data-Item -> SIP-Authentication-Scheme—always set to SIP_DIGEST
- Server-Name—the *home-server-route* parameter in the *sip registrar* configuration element. It is the URI (containing FQDN or IP address) used to identify and route to this Oracle CSM.

The Oracle CSM expects the MAA to include a SIP-Auth-Data-Item VSA, which includes digest realm, QoP and H(A1) information as defined in RFC2617. The information is cached for subsequent requests. Any result code received from the HSS other than DIAMETER_SUCCESS results in a 403 error response returned for the original request.

The MAR/MAA transaction is conducted with the server defined in the *credential retrieval config* parameter found in the *sip-authentication profile* configuration element. This parameter is populated with the name of a *home-subscriber-server* configuration element.

SIP Authentication Challenge

When the Oracle CSM receives a response from the HSS including the hash value for the user, it sends a SIP authentication challenge to the endpoint, if the endpoint did not provide any authentication headers in its initial contact with Oracle CSM. If the endpoint is registering, the Oracle CSM replies with a 401 Unauthorized message with the following WWW-Authenticate header:

```
WWW-Authenticate: Digest realm="atlanta.com",
domain="sip:boxesbybob.com", qop="auth",
nonce="f84f1cec41e6cbe5aea9c8e88d359", opaque="", stale=FALSE,
algorithm=MD5
```

If the endpoint initiates any other request to the Oracle CSM besides REGISTER, the Oracle CSM replies with a 407 Proxy Authentication Required message with the following Proxy-Authenticate header:

```
Proxy-Authenticate: Digest realm="atlanta.com", qop="auth",
nonce="f84f1cec41e6cbe5aea9c8e88d359", opaque="", stale=FALSE,
algorithm=MD5
```

Authentication Header Elements

- Domain—A quoted, space-separated list of URIs that defines the protection space. This is an optional parameter for the "WWW-Authenticate" header.
- Nonce—A unique string generated each time a 401/407 response is sent.
- Qop—A mandatory parameter that is populated with a value of "auth" indicating authentication.
- Opaque—A string of data, specified by the Oracle CSM which should be returned by the client unchanged in the Authorization header of subsequent requests with URIs in the same protection space.
- Stale—A flag indicating that the previous request from the client was rejected because the nonce value was stale. This is set to true by the SD when it receives an invalid nonce but a valid digest for that nonce.
- Algorithm—The Oracle CSM always sends a value of "MD5"

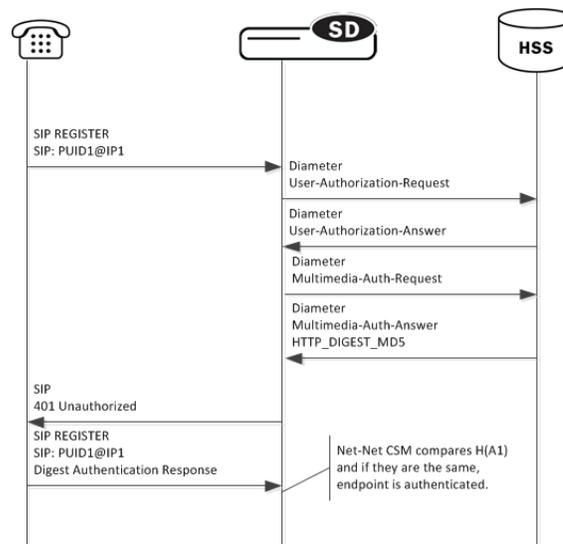
SIP Authentication Response

After receiving the 401/407 message from the Oracle CSM, the UA resubmits its original request with an Authorization: header including its own internally generated MD5 hash.

Oracle CSM Authentication Check

At this point, the Oracle CSM has received an MD5 hash from the HSS and an MD5 hash from the UA. The Oracle CSM compares the two values and if they are identical, the endpoint is successfully authenticated. Failure to match the two hash values results in a 403 or 503 sent to the authenticating endpoint.

The following image shows the User Authorization and Authentication process:



The Oracle CSM acts as a SIP Registrar and updates an HSS with the state of its registrants.

Oracle CSM as Registrar

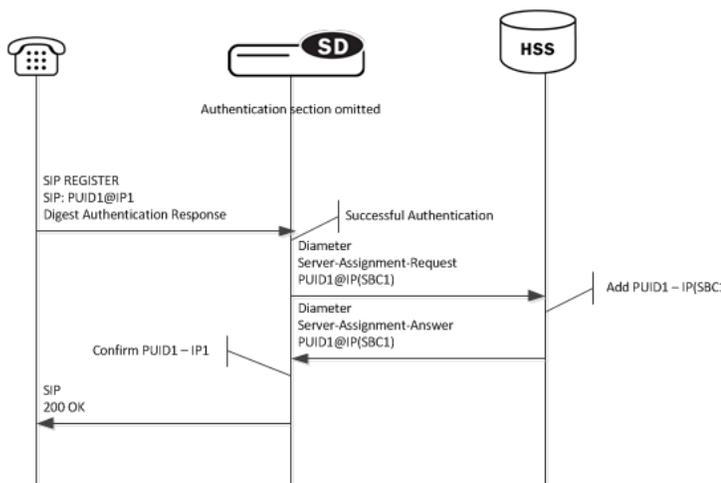
Creating a sip registrar configuration element enables the Oracle CSM to act as a SIP registrar. When registration functionality is enabled, the Oracle CSM actually registers endpoints rather than only caching and forwarding registrations to another device. Oracle CSM registry services are enabled globally per domain, not on individual SIP interfaces or other remote logical entities.

On receiving a REGISTER message, the Oracle CSM checks if it is responsible for the domain contained in the Request-URI as defined by the *domains* parameter and finds the corresponding *sip registrar* configuration. This is a global parameter—all messages are checked against all *sip registrar domains*. Thus you could create one sip registrar configuration element to handle all .com domains and one sip registrar configuration element to handle all .org domains. The Oracle CSM begins registrar functions for all requests that match the configured domain per sip-registrar configuration element.

A UA is considered registered once a SAA assignment is received from the HSS, after which the Oracle CSM sends a 200 OK message back to the registering UA.

New Registration

The following image shows a simplified call flow for a registering user:



Limiting AOR Contacts

The Oracle CSM allows you to limit the number of contacts that apply to AORs. If the Oracle CSM receives a registration request that exceeds the maximum that you configured, it responds with a local response, a 403 Forbidden by default, and does not register the additional contact. The system only rejects registration requests that exceed the maximum. Existing contacts persist normally.

The system checks against the maximum in the following circumstances:

- A new registration is received
- The location-update-interval expires
- A call-id changes (and the `forward-reg-call-id-change` option is enabled)
- A registrar message sequence number has skipped a number

- There is any change to the contact list

If the number of contacts in the initial registration exceeds the maximum, the Oracle CSM rejects the entire registration. In addition, if you configure this feature while the system is operational, your setting only applies to new registrations.

You configure these maximums on a per-registrar basis. The value ranges from 0-256. The feature is RTC supported.

HSS Server Assignment

As the Oracle CSM registers UAs, it requests to assign itself as the S-CSCF for the registering AoR. The Oracle CSM's S-CSCF identity is configured in the *home-server-route* parameter in *sip-registrar* configuration element. This is entered as a SIP URI (containing FQDN or IP address) and is used to identify and route messages to this Oracle CSM on behalf of the registered user.

Server Assignment Messages

The Oracle CSM sends a Server Assignment Request (SAR) to the HSS requesting to confirm the SIP or SIPS URI of the SIP server that is currently serving the user. The SAR message also serves the purpose of requesting that the Diameter server send the user profile to the SIP server. The SAR's AVPs are populated as follows:

- Public-User-Identity—the SIP AOR of the endpoint being registered (same as UAR)
- Private-User-Identity—the username from the SIP authorization header, if it is present. If not, this value is the public User ID. (Same as UAR)
- Server-Name—the *home server route* parameter in the *sip-registrar* configuration element. It is the FQDN or IP address used to identify and route to this Oracle CSM sent as a URL.
- Server-Assignment-Type—the value of this attribute depends upon the registration state:
 - REGISTRATION (1)—for all new and refreshing registrations.
 - Set to TIMEOUT_DEREGISTRATION (4)—when the contact is unregistered due to expiration. This occurs if the *force-unregistration* option is configured in the *sip config*.
 - USER_DEREGISTRATION (5)—when the contact is unregistered by the user (contact parameter expires=0).
- User-Data-Already-Available—always set to USER_DATA_ALREADY_AVAILABLE (1)

Server-Assignment-Response

The Oracle CSM expects a DIAMETER_SUCCESS code in the SAA to indicate that the assignment was successful. Then a 200 OK response is returned to the registering user. Any other Diameter result code is an error and results in an error response for the original REGISTER request (by default 503) and the contacts to be invalidated in the registration cache.

Register Refresh

When a UA sends a register refresh, the Oracle CSM first confirms that the authentication exists for that UE's registration cache entry, and then is valid for the REGISTER refresh. (If a valid hash does not exist for that AoR, then the Oracle CSM sends an MAR to the HSS to retrieve authentication data once again).

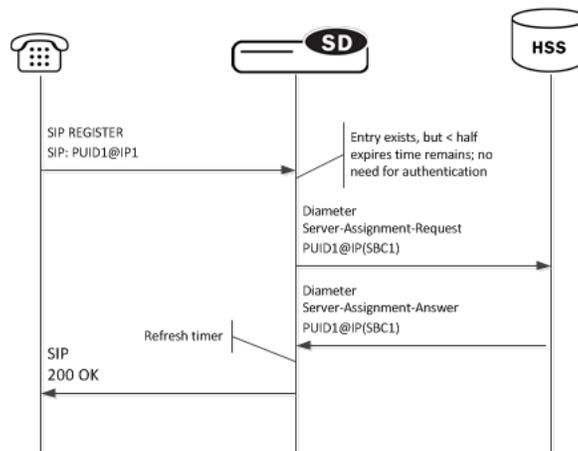
Next, the Oracle CSM determines if it can perform a local REGISTER refresh or if the HSS needs to be updated. If any of the following 3 conditions exists for the re-registering UA, the Oracle CSM updates the HSS:

1. The *location update interval* timer has expired—This value, configured in the *sip registrar* configuration element ensures that HSS database always has the correct Oracle CSM address by periodically sending SARs for each registered contact.
2. The message's call-id changes while the **forward-reg-callid-change** option in the *sip config* configuration element is set. This covers the case where the UA changes the Oracle CSMs through which it attaches to the network.
3. The REGISTER message's Cseq has skipped a number. This covers the case in which a user registered with Oracle CSM1, moves to Oracle CSM2, and then returns to Oracle CSM1.

If the Oracle CSM updates the HSS database because of matching one of the above conditions, the access side expiration timer per contact is reset to the REGISTER message's Expires: header value, and returned in the 200 OK. This happens even in the case when the reREGISTER was received in the first half of the previous Expires period. In addition, the core-side *location update interval* timer are refreshed on both active and standby.

When the above three conditions are not met, the registration expiration proceeds normally.

If the timer has not exceeded half of its lifetime, a 200 OK is returned to the UA. If the timer has exceeded half of its lifetime, the Oracle CSM just refreshes the access-side expiration timer; the registration cache expiration timer for that AoR begins its count again.



Core-side SAR Lifetime

The Oracle CSM maintains a timer for user registrations per SAR on the core side as specified above. The core-side SAR lifetime timer is configured in the *location update interval* parameter in the *sip registrar* configuration element. This timer ensures that the HSS always has the correct Oracle CSM address, by sending SAR messages periodically.

Entry Unregistration

Because AoRs and not contacts are referenced by the HSS, an AoR is valid and should not be removed from HSS until all associated contacts have been removed or

expired. If all the contacts are removed for an AoR by receiving REGISTER messages with `Expires:0` header, then the SAR sent to the HSS includes Server-Assignment-Type of USER_DEREGISTRATION (5).

When the *force-unregister* option in the *sip config* is enabled, then the HSS is explicitly updated when all of the contacts for an AoR have expired. This event prompts the Oracle CSM to send a SAR to the HSS using the Server-Assignment-Type of TIMEOUT_DEREGISTRATION (4).

The HSS can send a Registration-Termination-Request to request removing a registration, which corresponds to entries in the Oracle CSM's registration cache. When an RTR is received, the following AVPs are expected:

- Private-User-Identity—Username of the user, which is being de-registered.
- Associated-Identities—The Private-Id's in the same subscription which need to be de-registered. (optional)
- Public-Identity—One or more public-Id's of the user being de-registered. (optional)

For the AoR specified by the Private-User-Identity AVP, all associated contacts are removed in the registration cache. The Oracle CSM sends a Registration Termination Answer to the HSS to indicate success.

User Registration based on Reg-ID and Instance-ID (RFC 5626)

Sometimes a user's device reregisters from a different network than its original registration. This event should be considered a location update rather than a completely new registration for the Contact. The Oracle CSM can perform this way by considering the endpoint's reg-id and instance-id parameters defined in RFC 5626.

The Oracle CSM identifies new REGISTER requests received on a different access network as a location update of the existing binding between the Contact and AoR. Without this feature, the Oracle CSM would create a new binding and leave the old binding untouched in the local registration cache/ENUM database. This scenario is undesirable and leads to unnecessary load on various network elements including the Oracle CSM itself.

The following conditions must be matched to equate a newly registering contact as a location update:

For a received REGISTER:

1. The message must not have more than 1 Contact header while 1 of those Contact headers includes a reg-id parameter. (failure to pass this condition prompts the Oracle CSM to reply to the requester with a 400 Bad Request).
2. The Supported: header contains **outbound** value
3. The Contact header contains a **reg-id** parameter
4. The Contact header contains a **+sip.instance** parameter

After these steps are affirmed, the Oracle CSM determines if it is the First hop. If there is only one Via: header in the REGISTER, the Oracle CSM determines it is the first hop and continues to perform Outbound Registration Binding processing.

If there is more than 1 Via: header in the REGISTER message, the Oracle CSM performs additional validation by checking that a Path: header corresponding to the

last Via: includes an ob URI parameter, Outbound Registration Binding may continue.

If the Oracle CSM is neither the first hop nor finds an ob URI in Path headers, it replies to the UA's REGISTER with a 439 First Hop Lack Outbound Support reply.

reREGISTER Example

The user (AoR) bob@example.com registers from a device +sip.instance=<urn:uuid:0001> with a reg-id = "1", contact URI = sip:1.1.1.1:5060. A binding is created for bob@example.com+<urn:uuid:0001>+reg-id=1 at sip:1.1.1.1:5060.

Next, Bob@example.com sends a reREGISTER with the same instance-id but with a different reg-id = 2 and contact URI = sip:2.2.2.2:5060.

The previous binding is removed. A binding for the new contact URI and reg-id is created. bob@example.com+<urn:uuid:0001>+reg-id=2 at sip:2.2.2.2:5060

Outbound Registration Binding Processing

An outbound registration binding is created between the AoR, instance-id, reg-id, Contact URI, and other contact parameters. This binding also stores the Path: header.

Matching re-registrations update the local registration cache as expected. REGISTER messages are replied to including a Require: header containing the outbound option-tag.

If the Oracle CSM receives requests for the same AOR with some registrations with reg-id + instance-id and some without them, the Oracle CSM will store them both as separate Contacts for the AOR; The AoR+sip.instance+reg-id combination becomes the key to this entry.

Wildcarded PUID Support

The Oracle CSM supports the use of wildcarded Public User IDs (PUIDs), typically for registering multiple endpoints on a PBX with a single PUID. A wildcard is composed of a regular expression that, when used in a PUID prefix, represents multiple UEs. The group of UEs is referred to as an implicit registration set and share a single service profile. This support is typically implemented to reduce HSS resource requirements. The regular expressions themselves are in form of Perl Compatible Extended Regular Expressions (PCRE).

Each implicit registration set is associated with an explicitly registered "distinct" PUID. Typically, this distinct PUID is the PBX itself. The implicit registration set is dependent on the distinct PUID, including the distinct PUID's registration status.

There is no Oracle CSM configuration required.

Wildcarded PUID support is applicable to both I-CSCF and S-CSCF operation. In addition, all Oracle CSMs in the applicable data paths must be in the same trust domain.

To allow the feature, the Oracle CSM supports:

- Wildcarded PUID AVP in the LIR, SAR and SAA
- User Profile AVP in the SAA
- P-Profile-Key across the Mw interface, as defined in RFC 5002

Note also that the HSS must support the wildcarded-public-Identify AVP.

ACLI Instructions

The following configuration enables the Oracle CSM to authorize and authenticate registering users. In addition it sets the Oracle CSM to request itself as the S-CSCF for the registering users.

home subscriber server

To configure a home subscriber server (HSS):

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **home-subscriber-server** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# home-subscriber-server
ACMEPACKET(home-subscriber-server)#
```
4. **name**—Enter the name for this home subscriber server configuration element to reference from other configuration elements.
5. **state**—Set this to **enabled** to use this configuration element.
6. **address**—Enter the IP address of this HSS.
7. **port**—Enter the port which to connect on of this HSS, the default value is 80.
8. **realm**—Enter the realm name where this HSS exists.
9. Type **done** when finished.

SIP Authentication Profile

To configure the SIP Authentication Profile:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-authentication-profile** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-authentication-profile
ACMEPACKET(sip-authentication-profile)#
```

You may now begin configuring the SIP Authentication Profile configuration element.
4. **name**—Enter the name of this SIP authentication profile that will be referenced from a SIP registrar (or a SIP interface) configuration element.
5. **methods**—Enter all the methods that should be authenticated. Enclose multiple methods in quotes and separated by commas.
6. **anonymous-methods**—Enter the methods from anonymous users that require authentication. Enclose multiple methods in quotes and separated by commas.
7. **digest-realm**—Leave this blank for Cx deployments.
8. **credential-retrieval-method**—Enter CX.

9. **credential-retrieval-config**—Enter the *home-subscriber-server* name used for retrieving authentication data.
10. Type **done** when finished.

SIP Interface

The full SIP interface should be configured according to your network needs. Please refer to the Net-Net SBC ACLI Configuration Guide.

To configure a SIP Digest Authentication on a specific SIP Interface:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-interface** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-interface
ACMEPACKET(sip-interface)#
```
4. Type **select** and choose the number of the pre-configured sip interface you want to configure.


```
ACMEPACKET(sip-interface)# select
<realm-id>:
1: private 192.168.101.17:5060
2: public 172.16.101.17:5060

selection: 1
```
5. **registration-caching**—Set this parameter to **enabled**.
6. **ims-access**—Set this parameter to **enabled** for access interfaces, when applicable. Core interfaces should have this feature disabled.
7. **sip-authentication-profile**—Set this to the name of an existing *sip-authentication* profile if you wish to authenticate per SIP interface.
8. Type **done** when finished.

SIP Registrar

To configure the Oracle CSM to act as a SIP Registrar:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-registrar** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)#
```
4. **name**—Enter a name for this SIP registrar configuration element.
5. **state**—Set this to **enabled** to use this SIP registrar configuration element.
6. **domains**—Enter one or more domains that this configuration element will invoke SIP registration for. Wildcards are valid for this parameter. Multiple entries can be entered in quotes, separated by commas.

7. **subscriber-database-method**—Set this to **CX**.
8. **subscriber-database-config**—Enter the home-subscriber-server configuration element *name* that will handle REGISTER messages for this domain. The HSS configuration element includes the actual IP address of the server that SAR's are sent to.
9. **authentication-profile**—Enter a *sip-authentication-profile* configuration element's *name*. The *sip authentication profile* object referenced here will be looked up for a REGISTER message with a matching *domain* in the request URI. You may also leave this blank for the receiving SIP Interface to handle which messages require authentication if so configured.
10. **home-server-route**—Enter the identification for this Oracle CSM that will be sent as the Server-Name in MAR and SAR messages to the HSS. This value should be entered as a SIP URI.
11. **location-update-interval**—Keep or change from the default of 1400 minutes (1 day). This value is used as the timer lifetime for core-side HSS updates.
12. Type **done** when finished.

Maximum Number of Contacts

To configure a sip-registrar with a maximum of 10 contacts per AOR:

1. From superuser mode, use the following command sequence to access *sip-registrar* element.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)# select
```

Select the registrar you want to configure.

2. Specify the number of contacts.

```
AACMEPACKET(sip-registrar)# max-contacts-per-aor 10
AACMEPACKET(sip-registrar)# done
```

To configure local response for the Oracle CSM to issue when max-contacts-per-aor is exceeded:

1. From superuser mode, use the following command sequence to access *local-response* and add an entry.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# local-response-map
```

2. Access the entries configuration.

```
ACMEPACKET(local-response-map)# entries
```

3. Specify the local error you need to configure.

```
ACMEPACKET(local-response-map-entry)# local-error contacts-per-aor-exceed
```

4. Specify the sip-reason for this error.

```
ACMEPACKET(local-response-map-entry)# sip-reason forbidden
```

5. Specify the error code for this error.

```

ACMEPACKET(local-response-map-entry)# sip-status 403
ACMEPACKET(local-response-map-entry)# done
local-response-map-entry
    local-error                contacts-per-aor-exceed
    sip-status                  403
    q850-cause                  0
    sip-reason                  forbidden
    q850-reason
    method
    register-response-expires
ACMEPACKET(local-response-map-entry)# exit

```

SIP Registration Event Package Support

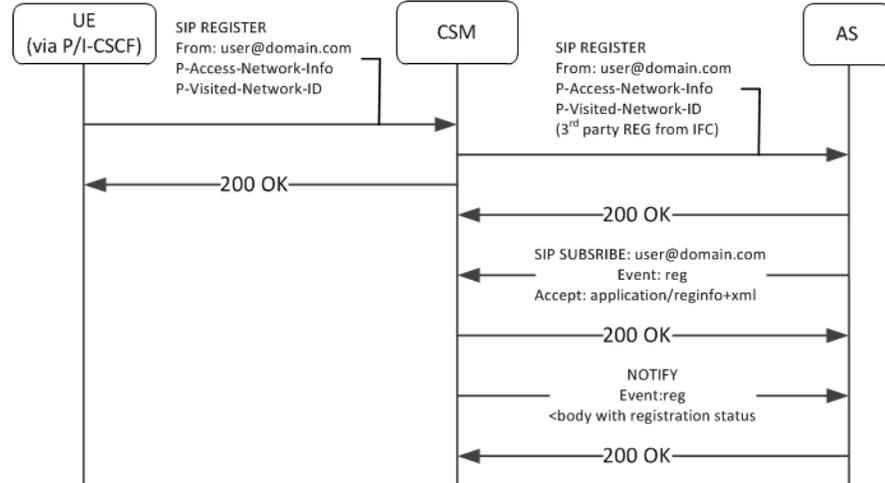
The Oracle CSM supports UA subscriptions to the registration event package, as defined in RFC3680. As such, it maintains contact with entities, often application servers, that need to know about UA registration events and provides those application servers with notifications when registration events occur.

Common usage for this functionality includes:

- Forcing re-authentication
- The provision of welcome notices to users who need information or instructions customized to their location

An operational example, shown below, begins with the Oracle CSM performing 3rd party registration on behalf of a UA to an AS, based on the iFC request from the UA. The AS, being an appropriately authorized UA itself, subscribes to NOTIFY messages on “reg” events for the initial UA. The Oracle CSM sends a 200OK to the

AS, and then proceeds to forward NOTIFY messages about that UE's registration events to the AS.



This feature is relevant when the Oracle CSM is performing S-CSCF functions. You enable this feature on the Oracle CSM per registrar, by simply creating a profile and applying it to the applicable registrar.

SUBSCRIBE Processing

When the Oracle CSM has the reg-event notification function enabled for a registrar, it includes the “allow-events” header in its 200OK replies to successful REGISTERS. This lets UEs know that they can subscribe to registration event packages.

When the Oracle CSM receives reg-event subscription requests, it follows the sequence below to process SUBSCRIBE requests for reg events:

1. Determines validity of the request.

Subscriptions cannot include more than one package name. If there is more than one package name in the request, the Oracle CSM replies with a 400 Bad Request message.

2. Determines if it can be a notifier, as follows:

- The SUBSCRIBE must include `EVENT=reg`.
- The requesting UA must be in the same domain as the registrar.

If both of the above are true, the Oracle CSM proceeds with the request.

3. Authorizes the request. The Oracle CSM only authorizes requests from UEs that come from the same realm and layer 2 connection on which it received the initial REGISTER.

Furthermore, the Oracle CSM only authorizes the following UEs:

- Public user identities from UEs that are subscribing to their own registration events.
- Public user identities that this user “owns”. Examples include implicitly registered public user identities.

- Entities that were included in the PATH header of the target UE's registration.
- All ASs that are listed in the UE's iFC and that are part of the trust domain.

If all of the above are true, the Oracle CSM proceeds with the request. If not, it sends 403 Forbidden to the requester.

4. Determines how it is functionally related to the UA. The Oracle CSM only processes subscriptions for users in its registration cache, replying with a 403 Forbidden if not. For cached users, the Oracle CSM forwards the request to the registrar if it is the P-CSCF. If it is the S-CSCF, it sends a 200 OK and begins to act as notifier.
5. Identifies the subscription duration, as follows, and sends the 200 OK to the UE:
 - If there is no Expires header in the UE's 200OK message, the Oracle CSM applies it's own configured minimum or the default (600000 seconds), whichever is greater.
 - If the SUBSCRIBE includes an Expires header, the Oracle CSM honors the request unless it is less than the configured minimum.
 - If the SUBSCRIBE's Expires header is less than the minimum subscription time configured in the registration event profile, the Oracle CSM denies the subscription, sending a 423 Too Brief message.

When the Oracle CSM encounters an Expires header set to 0, it terminates the subscription. This is referred to as "unsubscribing".

SUBSCRIBE REFRESH Requests

Subscriptions must be refreshed to keep them from expiring. ASs accomplish this by sending SUBSCRIBE REFRESH messages to the Oracle CSM. Messages must be received from authorized subscribers and on the same realm and connection as the original SUBSCRIBE or the Oracle CSM rejects the refresh request.

Reg Event NOTIFY Messages

When configured, the Oracle CSM issues NOTIFY messages to subscribed ASs when significant registration events occur. NOTIFY messages sent by the Oracle CSM comply fully with RFC3680. Events that trigger NOTIFY messages include:

- Registered
- Registration refreshed
- Registration expired
- Registration deactivated
- UE unregistered

The Oracle CSM does not send NOTIFY messages for the following events:

- Registration created
- Registration shortened
- Registration probation
- Registration rejected

Additional detail about NOTIFY messages that is specific to the Oracle CSM includes:

- The Oracle CSM always sends full information on all contacts, and indicates such within the `reginfo` element. The Oracle CSM does not utilize the partial state described within RFC 3680.
- Wildcarded PUIDs are included, enclosed in the `<wildcardedIdentity>` tag within the `<registration>` element.
- The Oracle CSM does not include the following optional attributes within the contact element:
 - expires
 - retry-after
 - duration registered
 - display-name
- The Oracle CSM uses the optional `unknown-param` element within the contact element to convey UA capabilities and distribute `reg-id`, `sip.instance` and `header filed` attributes.

An example of the XML body of a NOTIFY message below documents the registration status for the AOR `joe@example.com`.

```
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance version="0"
state="full">
  <registration aor="sip:joe@example.com" id="as9" state="active">
    <contact id="6" state="active" event="registered">
      <uri>sip:joe@pc887.example.com</uri>
    </contact>
    <contact id="7" state="terminated" event="expired">
      <uri>sip:joe@university.edu</uri>
    </contact>
  </registration>
</reginfo>
```

Use the `show registration` and `show sipd subscription` commands to display all information about each subscription.

Reducing NOTIFY Traffic

RFC 3265 stipulates that the Subscription server sends NOTIFY messages to all subscribers when a UA sends a registration refresh. This can generate excessive NOTIFY traffic. You, however, can mitigate this by configuring the Oracle CSM to limit notification traffic. By specifying the number of seconds between NOTIFY messages, you prevent the Oracle CSM from sending notifications upon events that do not generate a change in the registration database.

Database changes that trigger notifications when this option is configured include:

- The Cseq number of the REGISTER message increases by more than 1
- The call-ID changes
- A contact parameter changes
- The number of contacts changes

Upon expiry of this timer, the Oracle CSM sends out a NOTIFY for every registration event subscription. Note also that the Oracle CSM does not send the `cseq` attribute in the `CONTACT` element when this interval is configured.

Configuring Registration Event Package

This section shows you how to create reg-event profiles and apply those profiles to sip-registrars. These profiles enable the monitoring of UA registration events and the delivery of state change notifications to each UA that subscribes to the package. The procedure includes:

- Create one or more registration-event profiles
- Apply each profile to the applicable sip-registrar
- Optionally specify the registration event notification interval timer

ACLI Instructions

To configure a registration event profile:

1. From superuser mode, use the following command sequence to access *regevent-notification-profile* command.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# regevent-notification-profile
ACMEPACKET(registration-event-profile)#
```

2. To define the profile, simply name it and specify a “timeout” in seconds.

```
ACMEPACKET(registration-event-profile)# name reg-event-profile1
ACMEPACKET(registration-event-profile)# min-subscription-duration 2500
ACMEPACKET(registration-event-profile)# done
ACMEPACKET(registration-event-profile)# exit
```

3. Navigate to the registrar for which you want registration event package support.

```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)# regevent-notification-profile reg-event-profile1
ACMEPACKET(sip-registrar)# done
ACMEPACKET(sip-registrar)# exit
```

To specify optional NOTIFY refresh frequency:

1. From superuser mode, use the following command sequence to access *registration-event-profile* command within session router.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# regevent-notification-profile
ACMEPACKET(registration-event-profile)#
```

2. To enable NOTIFY, set the send-notify-for-reg-refresh option to the time, in seconds,

```
ACMEPACKET(registration-event-profile)# options notify-refresh-interval=1800
ACMEPACKET(registration-event-profile)# done
ACMEPACKET(registration-event-profile)# exit
```

Prepend the option with the + sign if you have multiple options configured that you want to retain.

```
ACMEPACKET(registration-event-profile)# options +notify-refresh-interval=1800
```

Running the command without the “+” character causes the system to remove any previously configured options.

Message Routing

The Oracle CSM performs routing in two ways depending on the *routing precedence* parameter in the *sip registrar*. Routing precedence can be set to either **registrar** (HSS) or **local policy**. Routing precedence is set to **registrar** by default.

Registrar routing uses the configured subscriber database and registration cache to route the call. Local policy routing lets you configure routing decisions within the Oracle CSM's local policy routing functionality.

Depending on whether the Oracle CSM is performing originating or terminating services for the call, described below, it performs those services prior to routing to the endpoint.

If, for any reason, the Oracle CSM is unable to proceed with routing a request, it replies to the UA that sent the request with a 4xx response.

Registrar Routing

When the *routing precedence* parameter is set to **registrar**, the Oracle CSM is using the HSS as a resource within the context of its routing decisions.

When an INVITE arrives, the Oracle CSM checks its own registration cache for a pre-existing matching contact in the INVITE. If it finds a match, it forwards the request to that location. If it does not find a match, it issues an Location Information Request (LIR) to the HSS. If the HSS's response, called an LIA, provides an assigned S-CSCF for that UA, the Oracle CSM proceeds as described below in the section LIR/LIA Transaction.

Note that you can configure the Oracle CSM to fallback to a local policy lookup if the lookup via the registrar fails. Configure this by adding the **fallback-to-localpolicy** option to the *sip-registrar* configuration element.

For situations where the database routing decision needs to be done in lieu of the default, you can set routing precedence to local-policy. Note that you can configure a routing entry that points to an HSS by setting a *policy attribute* with a next-hop of `cx:<home-subscriber-server-name>` within the *local-policy*.

LIR/LIA Transaction

An LIR includes the Public-User-Identity AVP, which contain a UA's actual PUID. The HSS responds with the assigned S-CSCF server (often a Oracle CSM) for this PUID. The answer is the form of a Location Info Answer (LIA). The LIA includes the assigned S-CSCF in the Server Name AVP.

If the S-CSCF returned in the LIR is *this* Oracle CSM, then the Oracle CSM performs unregistered termination services for this UA. (This situation indicates that the UA is currently unregistered.) Such services could include directing the call to voice mail. If the HSS returns an S-CSCF in the LIA that is not this Oracle CSM, it forwards the request to that S-CSCF.

Default Egress Realm

The *sip registrar* configuration element should be configured with a default *egress realm id*. This is the name of the *realm config* that defines the IMS control plane, through which all Oracle CSMs, HSSs, and other network elements communicate and exchange SIP messaging. It is advisable to configure this parameter to ensure well defined reachability among Oracle CSMs.

Routing Based on UA Capabilities

In compliance with RFC 3841, the Oracle CSM is able to make forwarding and forking decisions based on preferences indicated by the UA. To do this, the Oracle CSM evaluates each callee's AOR contact to determine the capabilities advertised by the UA and uses this information to make forwarding and forking decisions.

Prior to this support, the Oracle CSM made routing preference decisions solely via the "q" value present in the contact header. In cases where the preferences were equal, the Oracle CSM simply forwarded to those contacts simultaneously (parallel forking). In cases where the "q" value were not equal, the Oracle CSM forwarded in sequence (sequential forking), forwarding to the highest "q" value first.

The Oracle CSM now extends upon this functionality by scoring contacts, based on their capabilities, and making forwarding decisions using that score in addition to the "q" value.

There is no additional Oracle CSM configuration required to enable or invoke this processing. This functionality is supported for HSS, ENUM and Local Database configurations.

UE Capabilities

RFC2533 includes a framework that defines feature sets. Feature sets make up a group of media capabilities supported by a UA, individually referred to as media feature tags. In session networks, feature tag information is converted to a form specified in RFC3840 and exchanged between devices in the network to establish lists of UA capabilities. Based on these capabilities, session operation procedures are performed that facilitate preferred communications modalities.

RFC3840 defines:

- The format a UA uses to specify feature sets
- How a UA registers capabilities within the network
- An extension to the contact header so that it can include feature parameters
- The media tags that specify each capability

The full list of applicable media tags is presented in RFC 3840. Examples of tags include audio, automata, data, mobility, application and video.

Registering Capabilities at the Oracle CSM

Endpoints register their capabilities by listing them in the Contact headers of the REGISTER request. The Oracle CSM stores these feature parameters in its registration cache along with the other contact information. In the case of ENUM databases, the Oracle CSM also sends capabilities information to the ENUM infrastructure so that it can maintain capabilities records.

In addition to the standard set of tags, the Oracle CSM supports storing custom feature tags. Tags formatted with a + sign preceding the tag are recognized as custom tags. The exception to this are tags formatted using +sip.<tagname>, which are registered sip media feature tags.

An example of a contact header specifying audio, video and methods capabilities is shown below:

```
contact: sip:u1@h.example.com;audio;video;methods="INVITE,BYE";q=0.2
```

Preferential Routing

The Oracle CSM routes using UA capabilities only when acting as S-CSCF. It calculates preferred forwarding and forking using this information in conjunction

with UA requests. This calculation is based on Preferential Routing, as defined in RFC3841. Note that the “q” value is used in this calculation.

Using Preferential Routing, the Oracle CSM creates a target UA list from applicable contacts by matching capabilities with preferences. After creating the match list, it scores UEs based on how closely they match the preferred criteria. The system determines the forwarding order referring to the q value first and then the routing score. UEs for which both scores are equal are forwarded at the same time. All remaining UEs are forwarded sequentially.

The table below presents an example wherein the result of matching and scoring calculations causes the Oracle CSM to forwards sequentially to UE3, then UE2, then UE1.

User Agent	q Value	Preferential Score
UE3	1000	1000
UE1	500	1000
UE2	1000	700

UAs may or may not include capability request information in their messages. Preferential routing processing accounts for this by defining both *explicit* and *implicit* feature preference processing procedures.

Explicit Feature Preference

RFC3841 defines the two headers that a UA can use to explicitly specify the features the target UA must support, including:

Accept-Contact: — UEs the session initiator would like to reach

Reject-Contact: — UEs the session initiator does not want to reach

When the Oracle CSM receives messages that includes these headers, it gathers all the contacts associated with the AOR and creates a target list using preferential routing calculations. The example below, drawn from RFC 3841, specifies the desire to route to a mobile device that can accept the INVITE method.

Accept-Contact: `*;mobility="mobile";methods="INVITE"`

The “require” and “explicit” Feature Tag Parameters

RFC 3841 defines operational procedures based on the `require` and `explicit` feature tag parameters, which the Oracle CSM fully supports. UAs include these parameters in the `accept-contact:` header to further clarify capabilities requirements for the session. The Oracle CSM can use these parameters to exclude contacts or specify the forwarding order.

To summarize the use of these parameters per RFC 3841:

- When both parameters are present, the Oracle CSM only forwards to contacts that support the features and have registered that support.
- If only the `require` parameter is present, the Oracle CSM includes all contacts in the contact list, but uses a forwarding sequence that places the “best” match (with the most matching capabilities) first from those with the same “q” value.
- If only the `explicit` parameter is present, the Oracle CSM includes all contacts in the contact list, but uses a forwarding sequence that places contacts that have explicitly indicated matching capabilities before those

with the same “q” value. Unlike requests that specify both require and explicit, non-matching contacts may be tried if the matching ones fail.

- If neither parameter is present, the Oracle CSM includes all contacts in the contact list, but determines a “best” match based on the “closest” match to the desired capabilities. Again the forwarding order starts with contacts that have the same “q” value.

Note that this preferential routing sequence can proceed with attempts to reach contacts with a lower “q” value after the sequences above are exhausted. Note also that the orders calculated by preferential routing never override any forwarding order specified by the UA.

Implicit Feature Preference

If the caller does not include accept-contacts or reject-contacts in the message, the Oracle CSM makes *implicit* feature preference assumptions. Implicit feature preference forwards messages to target UEs that support the applicable method, and, in the case of SUBSCRIBE requests, that support the applicable event.

For implicit feature preference cases, the Oracle CSM uses the UE’s “q” value solely to determine parallel and sequential forking.

ACL Instructions

SIP Registrar

To configure a SIP registrar configuration element for message routing:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-registrar** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)#
```
4. Type **select** and choose the number of the pre-configured sip interface you want to configure.
5. **routing-precedence**— Set this to either **registrar** or **local-policy** depending on your deployment.
6. **egress-realm-id**—Enter the default egress realm for Oracle CSM messaging.
7. Type **done** when finished.

Home Subscriber Server

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **home-subscriber-server** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# home-subscriber-server
ACMEPACKET(home-subscriber-server)#
```

4. Type **select** and choose the number of the pre-configured sip interface you want to configure.
5. Type **done** when finished.

Tel-URI Resolution

The Oracle CSM can initiate number resolution procedures for requests that have tel-URI or SIP-URI (with user=phone) numbers in the R-URI. It does this by querying number resolutions services, including the local routing table(s) or ENUM server(s) to resolve the R-URI to a SIP URI. In addition, the original R-URI may not include a full E.164 number. As such, you can also configure the Oracle CSM to perform a number normalization procedure and ensure it presents a full E.164 number for resolution. Upon successful resolution, the Oracle CSM proceeds with ensuing signaling procedures.

To configure the Oracle CSM to perform these lookups, you create applicable `local-routing-config` or `enum-config` elements and set an option within the `sip-registrar` that specifies a primary and, optionally, a secondary `local-routing-config` or `enum-config` that the `sip-registrar` uses for LRT or ENUM lookups. If there is no ENUM configuration on the `sip-registrar`, the Oracle CSM forwards applicable requests to a border gateway function via local policy.

Refer to the Net-Net SD ACLI Configuration guide, Session Routing and Load Balancing chapter for complete information on how to configure a `local-routing-config` and an `enum-config`.

Number Lookup Triggers

Use cases that are applicable to number lookups and the associated Oracle CSM procedures include:

- Request from the access side
 1. The Oracle CSM performs originating services
 2. If the R-URI is a tel-URI or SIP-URI (with user=phone) and is not in the Oracle CSM cache, it requests e.164 resolution from the ENUM server(s).
- Request from core side including request for originating services
 1. The Oracle CSM performs originating services
 2. If the R-URI is a tel-URI or SIP-URI (with user=phone) and is not in the Oracle CSM cache, it requests e.164 resolution from the ENUM server(s).
- Request from core side, for terminating services only
 1. If the R-URI is a tel-URI or SIP-URI (with user=phone) and is not in the Oracle CSM cache, it performs an LIR.
 2. If the LIA reply indicates the tel-URI or SIP-URI (with user=phone) is not provisioned, the Oracle CSM requests e.164 resolution from the ENUM server(s).

Actions Based on Lookup Results

The Oracle CSM forwards to the resultant SIP-URI under the following conditions:

- The SIP-URI is in the Oracle CSM cache, in which case the Oracle CSM performs terminating services.
- The SIP-URI is not in the Oracle CSM cache, and the Oracle CSM is configured to service the returned domain.
In this case, the Oracle CSM performs the following:
 1. The Oracle CSM issues an LIR for the SIP-URI.
 2. The Oracle CSM forwards the message to the correct S-CSCF.
- The SIP-URI is not in the Oracle CSM cache, and the Oracle CSM is not configured to service the returned domain.
In this case, the Oracle CSM performs refers to local policy to forward the message via local policy.

The Oracle CSM may not be able to use the results of the ENUM lookup. Such cases include:

- The lookup at the ENUM servers fail.
- The ENUM response includes the carrier identification code “cic” parameter.
- The ENUM response provides a number portability dip indicator “npdi” or routing number “rn”.

Such cases result in the Oracle CSM sending the request to a gateway, either a BGCF or an MGCF set, based on the rules within the local policy database.

Primary and Secondary ENUM Configs

For the purpose of redundancy, the Oracle CSM allows you to configure these number lookups to use a backup resource in case the lookup at the primary fails. Such scenarios include losing contact with the primary ENUM/LRT server config (query time-out) and the entry is not found at the primary (LRT or ENUM).

ACLI Instructions**To apply primary and secondary number lookup resources to a sip-registrar:**

1. From superuser mode, use the following command sequence to access the *sip-registrar* element and select the registrar you want to configure.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)# select
```

2. Specify the resources to use with the options command.

Prepend the option with the “+” character if you have multiple options configured that you want to retain. Running the command without the “+” character causes the system to disable any previously configured options.

To specify primary and secondary ENUM servers:

```
ACMEPACKET(sip-registrar)# options +e164-primary-config=enum:<enum-config name>
ACMEPACKET(sip-registrar)# options +e164-secondary-config=enum:<enum-config name>
ACMEPACKET(sip-registrar)# done
```

To specify primary and secondary LRT resources:

```
ACMEPACKET(sip-registrar)# options +e164-primary-config=lrt:<lrt-config name>
```

```
ACMEPACKET(sip-registrar)# options +e164-secondary-config=1rt:<1rt-
config name>
ACMEPACKET(sip-registrar)# done
```

Bear in mind that an `enum-config` can reference multiple servers. When the Oracle CSM references an `enum-config`, queries follow the normal `enum-config` sequence, checking each referenced server in order. If the lookup is not successful at the primary, the Oracle CSM checks the servers in the registrar's `e164-secondary-config`.

In addition, each `enum-config` may refer to a different top-level-domain. This allows you to configure the Oracle CSM to successfully perform lookups within two domains.

HSS Initiated User Profile Changes

The Oracle CSM can receive Push Profile Request (PPR) messages from an HSS and update the state of the IMS User Profile and associated subscription information it has cached locally. The SIP digest authentication information can also be updated and reassociated with an AoR in case that has changed too. The Oracle CSM expects to receive the following AVPs in a PPR message.

- Private-User-Identity—the username, whose subscription/authentication data has changed.
- SIP-Auth-Data-Item—if present new authentication data is included here.
- User-Data—if present new User data is included here.
- Charging-Information—if present new charging information is included here.

The Oracle CSM replies to an HSS's PPR in a PPA message with the following AVPs:

- Result-Code—indicates Diameter base protocol error. Valid errors for in a PPA are:
 - DIAMETER_SUCCESS—The request succeeded.
 - DIAMETER_ERROR_NOT_SUPPORTED_USER_DATA—The request failed. The Oracle CSM SBC informs HSS that the received user information contained information, which was not recognized or supported.
 - DIAMETER_ERROR_USER_UNKNOWN—The request failed because the Private Identity is not found in Oracle CSM SBC.
 - DIAMETER_ERROR_TOO_MUCH_DATA—The request failed. The Oracle CSM SBC informs to the HSS that it tried to push too much data into the Oracle CSM SBC.
 - DIAMETER_UNABLE_TO_COMPLY—The request failed.
- Experimental-Result—indicates diameter application (3GPP/Cx) error if present.

Other Diameter Cx Configuration

Host and Realm AVP Configuration for Cx

You can configure the values sent in the origin-host, origin-realm and destination-host AVPs when the Oracle CSM communicates with a server over the Cx interface. Configure destination-host when you want to precisely specify the HSS with which these Cx exchanges take place.

The applicable configuration parameters are located in the home-subscriber-server configuration element. The parameters used to configured the AVPs are origin-realm, origin-host-identifier and destination-host-identifier. The AVPs are constructed as follows:

```
Origin Host AVP = <origin-host-identifier>.<origin-realm>
Origin Realm AVP = <origin-realm>
Destination Host AVP = <destination-host-identifier>.<destination-realm>
```

If the **origin-realm** is not configured, then the realm parameter in the home-subscriber-server configuration element will be used as the default. If **origin-host-identifier** is not configured, then the name parameter in the home-subscriber-server configuration element will be used as the default.

If these parameters are not configured, then the AVPs are constructed as follows:

```
Origin Host = <HSS Config name>.<HSS Config realm>.com
Origin Realm AVP = <HSS Config realm>
Destination Host = <HSS Config name>.<HSS Config realm>.com
```

ACLI Instructions

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **home-subscriber-server** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# home-subscriber-server
ACMEPACKET(home-subscriber-server)#
```
4. **origin-realm**—Set this to a string for use in constructing unique Origin Host and Origin Realm AVPs.
5. **origin-host-identifier**—Set this to a string for use in constructing a unique Origin Host AVP.
6. **destination-host-identifier**—Set this to a string for use in constructing a unique Destination Host AVP.
7. Save your work.

Initial Filter Criteria (IFC)

The Oracle CSM, acting as a S-CSCF, downloads a set of rules known as Initial Filter Criteria (IFC) from the HSS/AS. IFCs are downloaded over the Cx interface.

iFCs are a way for an S-CSCF to evaluate which ASs should be involved in the call flow for a given user agent (UA). iFCs are functionally defined by Boolean statements, whose component parts are expressed in XML; they reference the destination AS(s) where a desired service is provided.

IFC Evaluation

IFCs are evaluated as described in 3GPP TS 29.228. The Oracle CSM supports all tags found in the 3GPP initial filter criteria specifications. An IFC is evaluated until its end, after which the call flow continues as expected.

SIP Registration

When the Oracle CSM receives an authenticated REGISTER request from a served UA, it sends an SAR request to the HSS to obtain an SAA which includes iFCs associated with the UE's subscription. Within the context of registration, the Oracle CSM also manages third party registration procedures in conjunction with iFC exchanges or manually via the ACLI. These procedures are described in the Third Party Registration chapter.

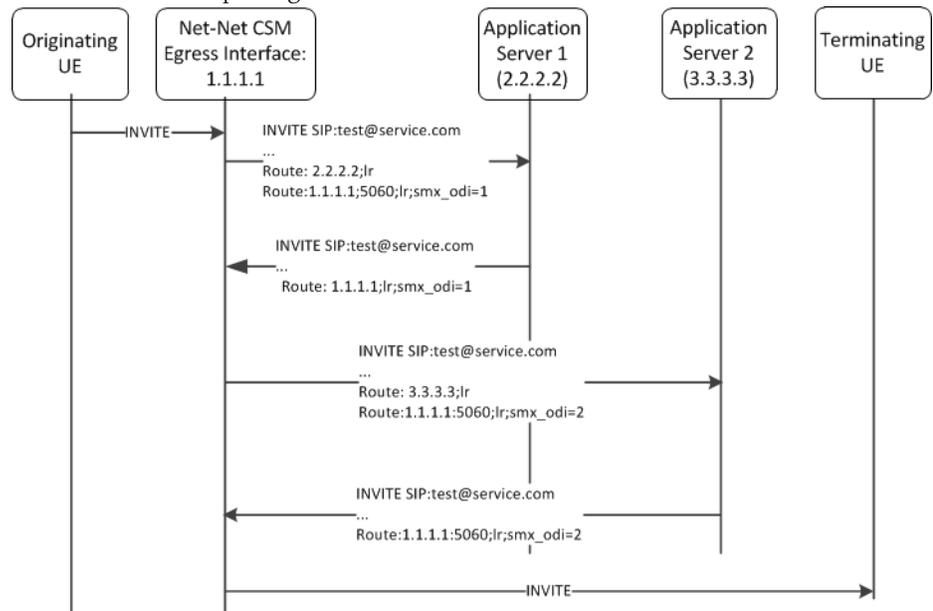
SIP Call

The Oracle CSM evaluates all iFC logic to determine that messages with matching characteristics are sent to the proper AS specified in the iFC evaluation using the IP Multimedia Service Control (ISC) interface. In this INVITE, the Oracle CSM adds two Route headers. The first (top) route header contains the target AS's URI. The second Route parameter is built using the IP address of the egress SIP interface and contains the ODI as a parameter. For example:

```
INVITE SIP:test@service.com
...
Route:2.2.2.2;lr
Route:1.1.1.1:5060;lr;smx_odi=1
```

If the AS routes the call back to the Oracle CSM, it is expected to include the ODI parameter that it received from the Oracle CSM, unchanged. The presence of the ODI parameter indicates that iFC evaluation needs to continue from where it left off for this call. If this continuation of iFC evaluation results in another AS URI, the Oracle CSM initiates a request towards that AS this time with a new ODI. In this way, the ODI is a state-signifier of Service Point Triggers.

The process continues until iFC evaluation is completed. Below is an example of an iFC evaluation completing after two iterations.



The iFC continues to be evaluated completely which may result in the INVITE being forwarded to additional ASs. At the conclusion of evaluating the iFC, the Oracle CSM checks if the target of the initial request is registered to itself, or not. If the UA is not registered locally the Oracle CSM forwards the request by regular means into the network. If the target UA is registered locally, the Oracle CSM proceeds to obtain iFCs for the target and begin iFC evaluation for the terminating side of the call.

Evaluating Session Case in the P-Served-User Header

The P-served-user header field conveys the identity of the served user, the session case that applies to the particular communication session, and application invocation, as defined in RFC 5502 and TS 24.229. The Session Case (sescase) and Registration State (regstate) parameters are either populated by the UA originating the message or by the Oracle CSM after it determines if a request is originating or terminating, and registered or unregistered.

The P-served-user header is created and added to an outgoing request if the next hop is trusted. A trusted next hop is an entity defined by a session agent whose trust-me parameter is enabled. Likewise, the P-served-user header is stripped if the request is forwarded to a next hop that is not known to be trusted.

When the Oracle CSM creates a P-served-User header, the user value in the originating case is the user value found in the P-Asserted-Identity header field. In the terminating case, the user value is taken from the Request URI.

Supported Sessioncase and Registration State

The following cases are supported for IFC evaluation. Conditions for classifying the calls as such are listed below.

Originating request by a UA, Registered User

When the Oracle CSM receives an Initial request, it is validated as an originating request from a registered user when the following conditions are met:

- The request is a dialog creating request or a standalone request.
- There is no "odi" parameter in the top route of the request.
- The regstate and sescase parameters of the P-served-user indicate for this to be treated as originating request for a registered user OR "The request is received from a registered contact.

Originating request by a UA, Unregistered User

When the Oracle CSM receives an Initial request, it is validated as an originating request from an unregistered user when the following conditions are met:

- The request is a dialog creating request or a standalone request.
- There is no "orig" parameter in the top route of the request.
- The served user is unregistered.
- The request is from an AS or I-CSCF and the top route header contains the "orig" parameter OR
The regstate and sescase of the P-served-user header indicates that the request is an originating request for an unregistered user.

Terminating Requests to a UA, Registered User

When the Oracle CSM receives an Initial request, it is validated as a terminating request towards a registered user when the following conditions are met:

- The request is a dialog creating request or a standalone request.
- There is no "orig" parameter in the top route of the request.
- There is no "odi" parameter in the top route of the request.
- The regstate and sescase parameters of the P-served-user indicate for this to be treated as terminating request for a registered user OR the request is finished

with originating services if applicable and the request is destined to a user who is currently registered with the Oracle CSM.

- If the Request-URI changes when visiting an application server, the Oracle CSM terminates the checking of filter criteria and routes the request based on the changed value of the Request-URI, per 3GPP Specification TS 23.218.

Terminating Requests to a UA, Unregistered User

See the following section [IFC Support for Unregistered Users \(38\)](#) for this case.

- If the Request-URI changes when visiting an application server, the Oracle CSM terminates the checking of filter criteria and routes the request based on the changed value of the Request-URI, per 3GPP Specification TS 23.218.

Additional Options

- The Oracle CSM can populate the top Route: header with the sescase value for ASs that require it. In such a case, the parameter is created as either call=orig or call=term. This behavior is enabled by configuring the **add-sescase-to-route** option in the *ifc-profile*.
- When the dialog-transparency parameter in the sip-config is set to enabled and your network includes multiple ASs, you should add the **dialog-transparency-support** option in the *ifc-profile*.

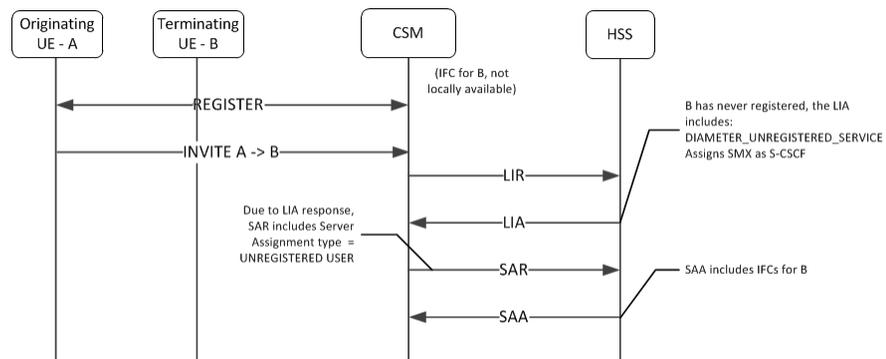
IFC Support for Unregistered Users

The Oracle CSM can download Initial Filter Criteria (IFC) from the HSS for unregistered users. This section displays applicable message sequence diagrams.

UE-terminating requests to an unregistered user

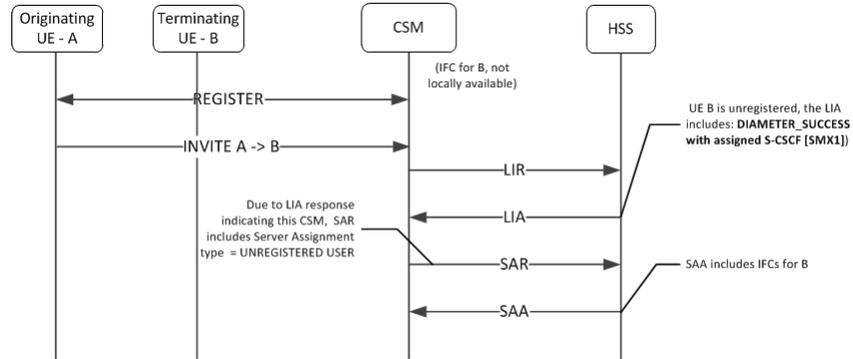
The Oracle CSM downloads and executes IFCs for the terminating end of calls. The following call flows indicate possible cases for the terminating unregistered user.

Terminating UA - Not Registered

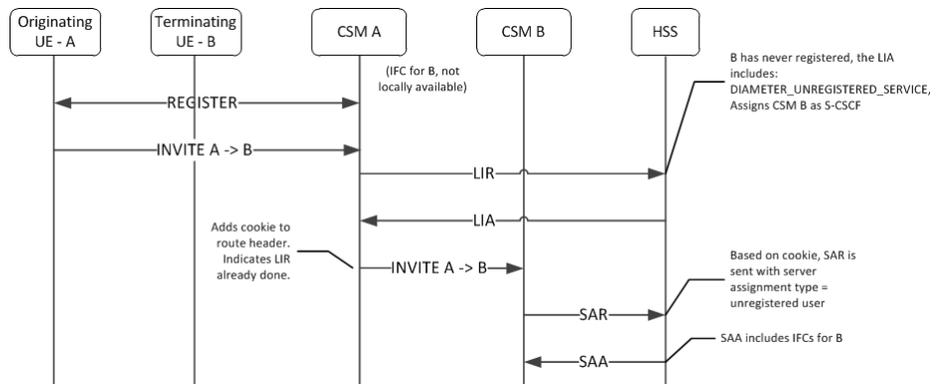


Terminating UA - Not registered

UE originally registered as a consequence of an originating or terminating request or an S-CSCF has stored the user profile.



Terminating UA - Not Registered, Served by other Oracle CSM



UE Subsequent Registration

If the Oracle CSM has a cached IFC downloaded for an unregistered UA who later registers to that Oracle CSM, the cached IFC will be cleared and updated with the IFC downloaded by the registration process.

Caching the Downloaded IFC

When the Oracle CSM downloads IFCs for unregistered users, they are saved to a local cache. If the IFC cache fills up, an older cached IFC for a user is released.

Optimizing IFC Updates

The Oracle CSM aims to reduce the number of IFC updates traversing the network to save bandwidth and transactional overhead. Unless the unregistered UE's IFC entry has been deleted because of exhausting cache space, the following optimizations are performed:

- If IFCs are available locally, then an SAR/SAA operation to download IFCs will not be performed.
- If a previous IFC download operation did not return any IFCs, then subsequent calls to that unregistered user will not invoke the SAR/SAA messaging to download IFCs.

Push Profile Request (PPR) updates

The HSS can push service profile updates for private IDs. The Oracle CSM can process PPR updates for unregistered entities. If the user entry has been deleted because IFC cache space has been exhausted, the PPRs will not be processed.

ACLI Instructions

SIP Registrar

To create an IFC Profile:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **ifc-profile** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# ifc-profile
ACMEPACKET(ifc-profile)#
```
4. **name**—Enter a name for this IFC profile.
5. **state**—Set this to enabled to use this ifc-profile.
6. **options**—Set the options parameter by typing options, a <Space>, the option name with a “plus” sign in front of it, and then press <Enter>.

If you type the option without the “plus” sign, you will overwrite any previously configured options. In order to append the new options to the options list, you must prepend the new option with a “plus” sign.

The options included in this section are: **add-sescase-to-route** and **dialog-transparency-support**.

7. Type **done** when finished.

SIP Registrar

To enable IFC support in a SIP Registrar:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-registrar** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)#
```
4. Type **select** and choose the number of the pre-configured SIP registrar configuration element you want to configure.


```
ACMEPACKET(sip-registrar)# select
name:
1: registrar1

selection:1
ACMEPACKET(sip-registrar)#
```
5. **ifc-profile**—Set this parameter to the name of the IFC profile you just created.

6. **servicing-function**—Set this parameter to disabled when you want the Oracle CSM to act solely as an I-CSCF. When disabled, the Oracle CSM always forwards requests from unregistered users to the servicing group. The default is enabled, which retains the S-CSCF function on this Oracle CSM.
7. **servicing-group**—Set this parameter to a Session Agent Group (SAG) name. The Oracle CSM forwards requests from unregistered users to this group when the servicing function parameter is disabled. Use of this parameter requires the prior configuration of a SAG that includes all prospective S-CSCFs. The name you give to that group is the name you specify as an argument to this parameter.
8. Type **done** when finished.

Shared and Default iFCs

The Oracle CSM supports Shared iFCs (SiFC), as defined by TS 29.229 and Default iFCs, which are an Acme Packet extension upon SiFCs. SiFCs provide an operator with the ability to create iFC templates and apply them to a large number of UEs. The SiFC process optimizes the provisioning, storage and transfer of service profile information. The default iFC (DiFC) establishes a configuration wherein the iFC associations are available on the Oracle CSM itself. This establishes a backup scenario in case the HSS is not responsive.

To support the SiFC feature on the Oracle CSM, you create a profile that refers to a local, XML-formatted file. This file specifies the iFCs to be shared. You apply these profiles to registrars to specify where they are used.

When an SiFC configuration is in place, the Oracle CSM notifies the HSS that it supports SiFCs within the Supported-Features AVP in the SAR. The HSS replies to confirm that it supports SiFCs within the SAA. The SiFC feature must be enabled on the HSS.

Note that the form and function of the SiFC and DiFC files are compatible. You can use the same file for both SiFC and DiFC configuration, if desired.

SiFC Usage

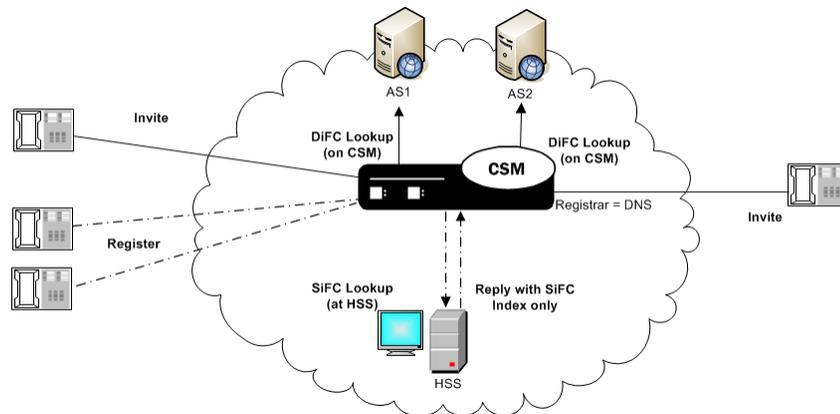
When an applicable end station registers, the Oracle CSM forwards the registration to the HSS normally. Given SiFC configuration however, the HSS sends a service-profile containing the SiFC identifiers to the Oracle CSM rather than the entire service definition. The Oracle CSM parses these identifiers and maps the user to the locally stored filter criteria.

The `<IFCSet id = "x">` tags in the XML file on the Oracle CSM map to the HSS identifiers.

DiFC Usage

In contrast to SiFCs, the Oracle CSM fires DiFCs within the context of a session. During the session, the Oracle CSM associates the iFCs within the DiFC file with the user, as needed. DiFC usage is invoked during session initiation.

Note that DiFCs are database agnostic. You can use DiFCs for HSS, ENUM and local database configurations. An operational overview of SiFCs and DiFCs is shown below.



SiFC/DiFC File Example

An example of a Oracle CSM local SiFC/DiFC XML file, including a single iFC Set containing a single iFC, is presented below.

```
<?xml version="1.0" encoding="UTF-8"?>
<IFCsets>
  <IFCSet id="0">
    <InitialFilterCriteria>
      <Priority>0</Priority>
      <TriggerPoint>
        <ConditionTypeCNF>0</ConditionTypeCNF>
        <SPT>
          <ConditionNegated>0</ConditionNegated>
          <Group>0</Group>
          <Method>INVITE</Method>
          <Extension></Extension>
        </SPT>
      </TriggerPoint>
      <ApplicationServer>
        <ServerName>sip:172.16.101.26:5060</ServerName>
        <DefaultHandling>0</DefaultHandling>
      </ApplicationServer>
      <ProfilePartIndicator>0</ProfilePartIndicator>
    </InitialFilterCriteria>
  </IFCSet>
</IFCsets>
```

Note that the Shared IFCSet contains the integer value property (`id="0"`) that associates these filter criteria with users registered with the Oracle CSM. In the case of SiFC, it is the value that the HSS should send when referencing shared sets. In the case of DiFC, the integer is meaningless. The Oracle CSM loads and executes default iFCs in the order they appear within the XML file.

iFC Execution Order

Within the context of the 3GPP standards, the Oracle CSM evaluates explicitly downloaded iFCs first when determining where to look for a service. If the Oracle CSM cannot execute on the service based on explicitly downloaded iFCs, it refers to the SiFC, then the DiFC information to identify an AS that supports the service.

Refreshing SiFC and DiFC Files

Given the nature of local file usage, an ACLI command is available to allow the user to refresh SiFC and DiFC contexts in memory after the user has saved changes to the SiFC and DiFC files. Run the following command to deploy these changes:

```
ACMEPACKET# refresh ifc <ifc-profile name>
```

Note also that the Oracle CSM validates the SiFC and DiFC files whenever you Activate your configuration.

ACLI Instructions

To configure the Oracle CSM to use Shared and Default IFCs:

- From superuser mode, use the following command sequence to access *ifc-profile* element.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# ifc-profile
```
- Define your profile.
- name**—Enter a name for this IFC profile.


```
ACMEPACKET(ifc-profile)# name acmeTelecomIFC
```
- state**—Set this to enabled to use this ifc-profile.


```
ACMEPACKET(ifc-profile)# state enabled
```
- default-ifc-filename**—Specify filename and, if not stored in the default directory /code/ifc, the applicable pathname.


```
ACMEPACKET(ifc-profile)# default-ifc-filename Afile.xml.gz
```
- shared-ifc-filename**—Specify filename and, if not stored in the default directory /code/ifc, the applicable pathname.


```
ACMEPACKET(ifc-profile)# shared-ifc-filename Bfile.xml.gz
```
- options**—Set the options parameter by typing options, a <Space>, the option name with a “plus” sign in front of it, and then press <Enter>.


```
ACMEPACKET(ifc-profile)# done
```
- Apply the ifc-profile to your sip registrar.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# sip-registrar
```

Select the registrar you want to configure and apply the profile.

```
ACMEPACKET(sip-registrar)# select
ACMEPACKET(sip-registrar)# ifc-profile acmeTelecomIFC
ACMEPACKET(sip-registrar)# done
```

Distinct and Wildcarded Public Service Identity (PSI) Support

The Oracle CSM supports the use of distinct Public Service Identity (PSI) and wildcarded PSIs, typically for specifying access to a service. There is no configuration required on the Oracle CSM to enable this support.

Administrators use individual PSI entries and/or wildcarded PSIs as service identifiers on an HSS. These identifiers provide the information needed to direct applicable messages to applicable application servers. Distinct PSIs can reside within individual PSI entries; wildcarded PSI entries are managed within iFC lists. Wildcarded PSI support is typically implemented to reduce HSS resource requirements. By configuring a wildcarded PSI, administrators can use a single record within the iFC to manage multiple resources.

A wildcard is composed of an expression that, when used in a user part, provides for access to multiple service resources. The regular expressions themselves are in form of Perl Compatible Extended Regular Expressions (PCRE).

For example, consider the following two service resources:

- sip:chatroom-12@core.com
- sip:chatroom-64@core.com

These two service resources can be represented simultaneously at the HSS using the following syntax:

- sip:chatroom-!.*!@core.com

The Oracle CSM caches filter criteria information that uses this wildcard syntax. This avoids the need for SAR/SAA exchanges between the Oracle CSM and the HSS every time an entity requests the service. The Oracle CSM is equally capable of caching distinct PSIs, which similarly offloads the need for SAR/SAA exchanges during service resource location processes.

For most call flows, the Oracle CSM does not evaluate the expression for the purpose of finding a match. Instead, it keeps the syntax provided by the HSS in its cache and provides the wildcarded syntax in the applicable AVP.

To allow the feature, the Oracle CSM supports:

- Wildcarded public user identity AVP in the LIA, SAR and SAA
- User Profile AVP in the SAA
- P-Profile-Key across the Mw interface, as defined in RFC 5002

Configuring SIP Ping OPTIONS Support

You can configure the Oracle CSM to respond to SIP ping OPTIONS. This support is typically configured on an S-CSCF so it can respond to pings OPTIONS sent by a P-CSCF:

ACLI Instructions

To configure an SIP Options Ping response support:

1. From superuser mode, use the following command sequence to access *ping-response* command on a sip-interface element.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
```

```
ACMEPACKET(session-router)# sip-interface
```

```
ACMEPACKET(sip-interface)# sel
```

2. Enable the support with the `ping-response` command.

```
ACMEPACKET(http-config)# ping-response enabled
```

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

ping-response—Enable `ping-response` to allow your device to respond to ping OPTIONS. For example, this feature is useful within hybrid deployment environments on a P-CSCF as a means of verifying the S-CSCF's availability. This configuration allows the S-CSCF to respond to SIP ping OPTIONS.

Redundancy and Load Balancing with HSS Servers

The Oracle CSM allows you to operate with multiple HSS servers, supporting:

- Redundancy - Continue normal operation despite HSS failure.
- Load Balancing - Divide the traffic load between HSS servers in a group of HSSs. Preference is based on the HSS list order configured on the Oracle CSM.

You configure HSS servers within HSS Groups to support this functionality. For redundancy, you create and assign HSS groups, and apply either the *hunt* or *fail-over* strategy to your HSS group. To implement load balancing, you configure the applicable HSS group with a the *round-robin* server allocation strategy. This functionality assumes the HSS infrastructure itself is configured for redundancy.

The Oracle CSM establishes and manages multiple Cx connections with each applicable HSS. This management is achieved by connection identifiers on the Oracle CSM that allow it to distinguish between connections. This provides the network with the flexibility of being able to use multiple paths to a given HSS regardless of AVP values.

About HSS Groups

You configure HSS groups based on your redundancy and failover design. You accomplish this by configuring your HSS groups with the applicable HSS servers. You then assign your group to a registrar. HSS group configuration does not preclude assigning an HSS in the group to a registrar individually.

HSS groups can contain individual HSSs. Members of an HSS group are prioritized by the server list; the first server in the list takes the highest priority; the last takes the lowest. You can manually disable an HSS group, if desired, which prevents the Oracle CSM from attempting to access any of the HSS servers via that group.

HSS group members do not need to reside in the same domain, network, or realm. The Oracle CSM can allocate traffic among member HSSs regardless of their location. It uses the allocation strategies you configure for the group to distribute traffic across the group members.

Group allocation strategies define how the Oracle CSM selects an HSS. For example, the hunt strategy selects HSSs in the order in which they are listed. Allocation strategies include the following:

Allocation Strategy	Description
failover	For HSS redundancy deployments, the failover strategy specifies that the Oracle CSM selects the next highest priority HSS server for all operations if the first HSS fails. The Oracle CSM does not resume operation with the initial HSS when it comes back into service.
hunt	For HSS redundancy deployments, the hunt strategy specifies that the Oracle CSM select HSSs in the order in which they are configured in the HSS group. If the first HSS is available, all traffic is sent to the first HSS. If the first HSS is unavailable, all traffic is sent to the second HSS. The system follows this process for all HSS servers in the group. When a higher priority HSS returns to service, all traffic is routed back to it.
roundrobin	This strategy targets HSS load balancing deployments. The Oracle CSM selects each HSS in the order in which it appears in the group list, routing diameter requests to each HSS in turn.

Paths taken by specific messaging is constrained by the purpose of that messaging, and refined by a group's allocation strategy. Applicable messaging includes UAR/UA, MAR/MAA, SAR/SAA and LIR/LIA. For both failover and hunt strategies, all messaging is sent to the current active server. For the round-robin strategy, messaging is distributed to group members sequentially, using the member list order.

Connection Failure Detection

The Oracle CSM detects that a connection between itself and a given HSS has failed if either a diameter request fails or the diameter DWR/DWA handshake fails. If the HSS does not respond to five requests, the Oracle CSM marks that HSS as out of service.

The Oracle CSM forwards unacknowledged messages to subsequent HSSs based on strategy. It changes the destination host AVP of these messages and marks them with the "T" flag. The HSS recognizes the "T" flag as an indication that the request may be a duplicate, caused by a problem in the network.

Periodically, the Oracle CSM attempts to establish diameter connections with out of service HSS servers. When those connections succeed, the Oracle CSM marks the HSS as in-service and resumes using it within the context of the configured redundancy and load balancing strategy.

Configuring HSS Groups

To configure HSS groups:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the system-level configuration elements.


```
ACMEPACKET(configure)# session-router
```

3. Type **hss-group** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.

```
ACMEPACKET(session-router)# hss-group
ACMEPACKET(hss-group)#
```

4. **name**—Enter a unique name for the HSS group in Name format.
5. **state**—Enable or disable the HSS group on the Oracle CSM. The default value is **enabled**. Valid values are:
 - enabled | disabled
6. **origin-host-identifier**— Set this to a string for use in constructing a unique Origin Host AVP. This setting always takes precedence over the origin-host-identifier configured for the home-subscriber-server. This setting is required.
7. **strategy**—Indicate the HSS server allocation strategy you want to use. The strategy you chose selects the HSS servers that will be made available by this **hss-group**. The default value is **hunt**. The valid values are:
 - **hunt**—Selects HSS servers in the order in which they are listed. For example, if the first server is online, all traffic is sent to the first server. If the first server is offline, the second server is selected. If the first and second servers are offline, the third server is selected. When the Oracle CSM detects that a higher priority HSS is back in service, it routes all subsequent traffic to that HSS.
 - **roundrobin**—Selects each HSS server in the order in which they are listed in the destination list, selecting each server in turn, one per session.
 - **failover** — Selects the first server in the list until failure is detected. Subsequent signaling goes to the next server in the list.
8. **hss-configs**—Identify the HSS servers available for use by this **hss-group**. This list can contain as many HSS servers as is necessary. An **hss-config** list value must correspond to a valid **hss-config**.

Display syntax for the **hss-configs** parameter by typing the question mark character after the parameter name on the ACLI.

```
ACMEPACKET(hss-group)# hss-configs ?
```

```
<string>  list of home-subscriber-server configs for this group
           for single-entry: hss1
           for multi-entry: (hss1 hss2)
           for adding an entry to an existing list: +hss3
           for deleting an entry from an existing list: -hss3
           for multiple entries add/remove from a list: +/- (hss1 hss2)
```

The following example shows an HSS group using the hunt allocation strategy applied.

```
hss-group
  name                group-test1
  state               enabled
  origin-host-identifier
  strategy            hunt
  hss-configs         hss1, hss2
  last-modified-by    admin@console
  last-modified-date  2013-05-13 14:58:01
```

The Oracle CSM contacts an ENUM server via DNS for two purposes:

- to obtain authentication data for a registering UA (often referred to as a UE)
- to query the user subscriber database (ENUM) regarding the registration state of the AoR and update the database with the latest information

Message Authentication for SIP Requests

The Oracle CSM authenticates requests by configuring the *sip authentication profile* configuration element. The *name* of this configuration element is either configured as a parameter in the *sip registrar* configuration element's *authentication profile* parameter or in the *sip interface* configuration element's *sip-authentication-profile* parameter. This means that the Oracle CSM can perform SIP digest authentication either globally, per domain of the Request URI or as received on a SIP interface.

After naming a *sip authentication profile*, the received methods that trigger digest authentication are configured in the *methods* parameter. You can also define which anonymous endpoints are subject to authentication based on the request method they send to the Oracle CSM by configuring in the *anonymous-methods* parameter. Consider the following three scenarios:

1. By configuring the *methods* parameter with REGISTER and leaving the *anonymous-methods* parameter blank, the Oracle CSM authenticates only REGISTER request messages, all other requests are unauthenticated.
2. By configuring the *methods* parameter with REGISTER and INVITE, and leaving the *anonymous-methods* parameter blank, the Oracle CSM authenticates all REGISTER and INVITE request messages from both registered and anonymous endpoints, all other requests are unauthenticated.
3. By configuring the *methods* parameter with REGISTER and configuring the *anonymous-methods* parameter with INVITE, the Oracle CSM authenticates REGISTER request messages from all endpoints, while INVITES are only authenticated from anonymous endpoints.

Credential Retrieval

The Oracle CSM requests authentication information from an ENUM server via DNS when it receives a REGISTER or other message from an endpoint. This server, which provides authentication information, is defined on the Oracle CSM in an *enum-config* configuration element that includes an *enum-servers* (the IP addresses of the servers) and *realm* parameters. Together, these two parameters define the DNS/ENUM server(s) which provide authentication data.

The target ENUM server is determined first by setting the *credential retrieval config* parameter to **enum-config** so the Oracle CSM will reference that *enum config*. Next, set the *credential retrieval config* parameter to the name of an *enum config* configuration element which is populated with the ENUM servers' IP addresses.

User Authentication Query

As soon as a request is received on a SIP interface and has been determined to require authentication, the Oracle CSM attempts to authenticate the endpoint. It sends a DNS TXT query including the UA's AoR to an ENUM database and expects the H(A1) defined in RFC2617 for the user being authenticated.

SIP Digest User Authentication

SIP Authentication Challenge

When the Oracle CSM receives a response from the ENUM server including the hash value for the user, it sends a SIP authentication challenge to the endpoint, if the endpoint did not provide any authentication headers in its initial contact with the Oracle CSM. If the endpoint is registering, the Oracle CSM replies with a 401 Unauthorized message with the following WWW-Authenticate header:

```
WWW-Authenticate: Digest realm="atlanta.com",
domain="sip:boxesbybob.com", qop="auth",
nonce="f84f1cec41e6cbe5aea9c8e88d359", opaque="", stale=FALSE,
algorithm=MD5
```

If the endpoint initiates any other request to the Oracle CSM besides REGISTER, the Oracle CSM replies with a 407 Proxy Authentication Required message with the following Proxy-Authenticate header:

```
Proxy-Authenticate: Digest realm="atlanta.com", qop="auth",
nonce="f84f1cec41e6cbe5aea9c8e88d359", opaque="", stale=FALSE,
algorithm=MD5
```

Authentication Header Elements

- **Digest Realm**—This value is configured in the *digest-realm* parameter in the *sip-registrar* configuration element. This parameter is mandatory when using the "ENUM-TXT" credential retrieval method.
- **Domain**—A quoted, space-separated list of URIs that defines the protection space. This is an optional parameter for the "WWW-Authenticate" header.
- **Nonce**—A unique string generated each time a 401/407 response is sent.
- **Qop**—A mandatory parameter that is populated with a value of "auth" indicating authentication.
- **Opaque**—A string of data, specified by the Oracle CSM which should be returned by the client unchanged in the Authorization header of subsequent requests with URIs in the same protection space.
- **Stale**—A flag indicating that the previous request from the client was rejected because the nonce value was stale. This is set to true by the SD when it receives an invalid nonce but a valid digest for that nonce.
- **Algorithm**—The Oracle CSM always sends a value of "MD5"

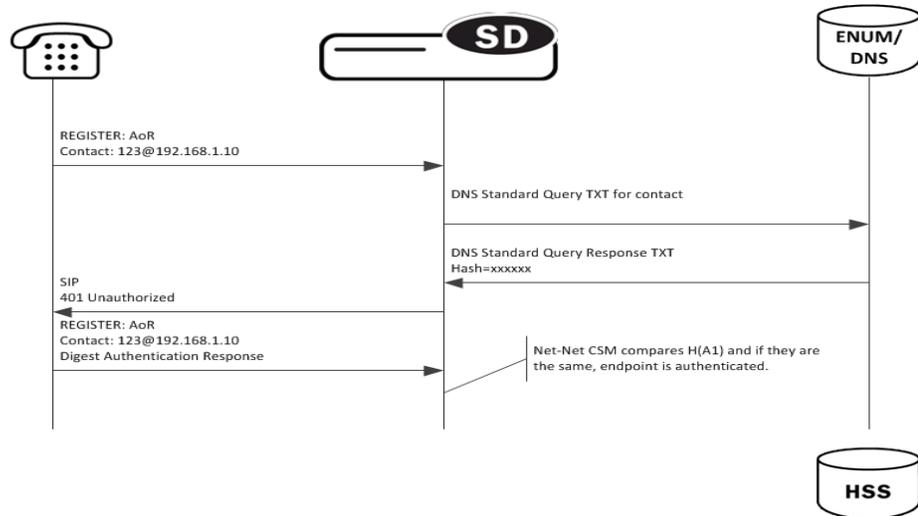
SIP Authentication Response

After receiving the 401/407 message from the Oracle CSM, the UA resubmits its original request with an Authorization: header including its own internally generated MD5 hash.

Oracle CSM Authentication Check

At this point, the Oracle CSM has received an MD5 hash from the ENUM server and an MD5 hash from the UA. The Oracle CSM compares the two values and if they are identical, the endpoint is successfully authenticated. Failure to match the two hash values results in a 403 or 503 sent to the authenticating endpoint.

The following image shows the user authentication process.



Oracle CSM as Registrar

DDNS Update to User Subscriber Database

As REGISTER messages are received, the Oracle CSM updates the ENUM database via DDNS UPDATE messages as defined by RFC2136. New registrations are added to the database while expired or deleted registrations are removed.

The Oracle CSM acts as a registrar by configuring the *sip registrar* configuration element. When registrar functionality is enabled, the Oracle CSM acts as a registrar rather than only caching and forwarding registrations to another device. Oracle CSM registry services are enabled globally per domain, not on individual SIP interfaces or other remote logical entities.

On receiving a REGISTER message, the Oracle CSM checks if it is responsible for the domain contained in the Request-URI as defined by the *domains* parameter and finds the corresponding *sip registrar* configuration. This is a global parameter and all messages are checked against all *sip registrar domains*. Thus you could create one sip registrar configuration element to handle all *.com domains and one sip registrar configuration element to handle all *.org domains. The Oracle CSM begins registrar functions for all requests that match the configured domain per sip-registrar configuration element.

A UA is considered registered after the Oracle CSM updates the ENUM server with a DDNS dynamic update. After this action, the Oracle CSM sends a 200 OK message back to the registering UA.

TTL

Part of the Oracle CSM architecture includes a local ENUM cache which maintains the results from ENUM queries locally. To enable Oracle CSM, the TTL value in the enum config must be set to 0. This ensures that whenever an ENUM query is required, the Oracle CSM consults the central User Subscriber Database to have the latest network-wide information about the UA it is trying to reach, instead of its ENUM cache.

ENUM Database Correlation

When a UA registers, as the number of associated contacts for an AoR grows or shrinks, the ENUM-based User Subscriber Database is updated in turn with the latest information using a DDNS UPDATE. After a REGISTER including authentication information is received from a UA, the Oracle CSM sends a Standard NAPTR query for the AoR to the ENUM server. The ENUM server replies with a Standard Query response, including the NAPTR records.

After receiving the entries from the ENUM database via the ENUM query, the list must be correlated with the Oracle CSM's view of the AoR's registration state. The differences will be resolved by sending a DDNS UPDATE to add or remove entries from the ENUM server. The database correlation phase only occurs when endpoints register.

Contacts that need to be added are put on the UPDATE add list. Contacts that are being unregistered (contact with Expires=0) or have expired timestamp are added to the UPDATE remove list.

Entry Expiration

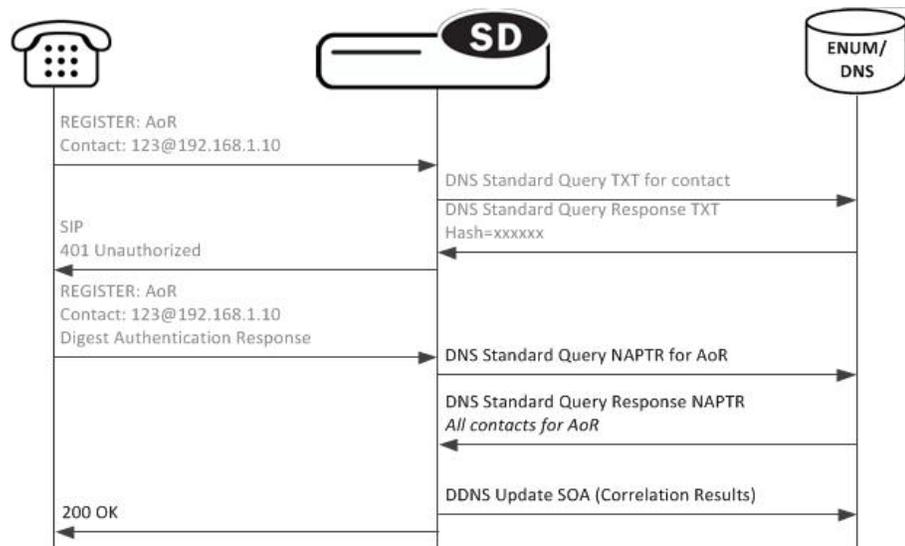
The Oracle CSM employs a process to remove expired contacts from the ENUM database whether they were entered by the active Oracle CSM or other Oracle CSM. After determining that a contact is expired, the Oracle CSM removes the contact in a subsequent DDNS update.

To do this, the Oracle CSM includes an expiration parameter in the Contacts it insert into the ENUM database. Expiration is indicated with a ts= parameter. This parameter's value is set to the initial registration time measured on the Oracle CSM plus the REGISTER message's Expires: header value or Expires parameter in the Contact header value. This value is measured in seconds after the epoch. In a DDNS update, a ts= parameter appears as follows:

Regex: "!^.*\$!sip:234-hchse6c0d01u2@172.16.101.51:5060;ts=1313493824!"

After the Oracle CSM retrieves contacts for an AoR in a NAPTR record that are expired, based on ts= parameters, the Oracle CSM's next Dynamic update tells the server to remove those contacts from the ENUM database.

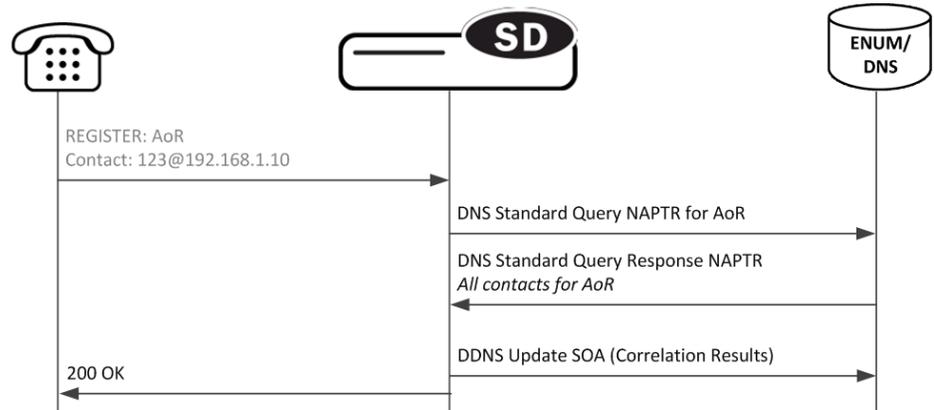
In this way, all Oracle CSM members of a domain may remove expired contacts from the ENUM database.



Register Refresh

When a UA sends a register refresh, the Oracle CSM first confirms that the authentication exists for that UA's registration cache entry, and then is valid for the REGISTER refresh. Then, the lifetime timer (value in Expires: header) for that registration cache entry is checked.

If the timer has not exceeded half of its lifetime, only a 200 OK is sent back to the UA. If the timer has exceeded half of its lifetime, the Oracle CSM sends a NAPTR update to the ENUM database.



In addition to the baseline Net-Net SBC REGISTER refresh conditions, an ENUM database update is required when one of the following conditions is satisfied:

1. The *location update interval* timer has expired—This value, configured in the *sip registrar* configuration element ensures that that ENUM database always has the latest user information by periodically sending Standard Queries.
2. The message's call-id changes while the **forward-reg-callid-change** option in the *sip config* configuration element is set. This covers the case where the UA changes the Oracle CSM through which it attaches to the network.
3. The REGISTER message's Cseq has skipped a number. This covers the case in which a user registered with Oracle CSM1, moves to Oracle CSM2, and then returns to Oracle CSM1.
4. The REGISTER message's contact list has changed.

After receiving the entries from the ENUM database via the NAPTR query, the list is correlated with the internal registration cache. Appropriate DDNS updates are preformed (see: [ENUM Database Correlation \(51\)](#)).

If the Oracle CSM updates the ENUM database due to one of the above conditions, the UA's access-side Expires timer is reset to the REGISTER message's Expires: header value, and returned in the 200 OK. This happens even in the case when the reREGISTER was received in the first half of the previous Expires period. In addition, the core-side *location update interval* timer is refreshed on both active and standby.

When the above four conditions are not met, the registration and reregistration expiration proceeds normally. If the access-side expiration timer has not exceeded half of its lifetime, only a 200 OK is sent back to the UA. If the timer has exceeded half of its lifetime, the Oracle CSM refreshes the registration to the ENUM server.

Note: Upon a Call-id or contact list change, both the registration cache timer and the ENUM database are updated.

Limiting AOR Contacts

The Oracle CSM allows you to limit the number of contacts that apply to AORs. If the Oracle CSM receives a registration request that exceeds the maximum that you configured, it responds with a local response, a 403 Forbidden by default, and does not register the additional contact. The system only rejects registration requests that exceed the maximum. Existing contacts persist normally.

The system checks against the maximum in the following circumstances:

- A new registration is received
- The location-update-interval expires
- A call-id changes (and the `forward-reg-callid-change` option is enabled)
- A registrar message sequence number has skipped a number
- There is any change to the contact list

If the number of contacts in the initial registration exceeds the maximum, the Oracle CSM rejects the entire registration. In addition, if you configure this feature while the system is operational, your setting only applies to new registrations.

You configure these maximums on a per-registrar basis. The value ranges from 0-256. The feature is RTC supported.

User Registration based on Reg-ID and Instance-ID (RFC 5626)

Sometimes a user's device reregisters from a different network than its original registration. This event should be considered a location update rather than a completely new registration for the Contact. The Oracle CSM can perform this way by considering the endpoint's reg-id and instance-id parameters defined in RFC 5626.

The Oracle CSM identifies new REGISTER requests received on a different access network as a location update of the existing binding between the Contact and AoR. Without this feature, the Oracle CSM would create a new binding and leave the old binding untouched in the local registration cache/ENUM database. This scenario is undesirable and leads to unnecessary load on various network elements including the Oracle CSM itself.

The following conditions must be matched to equate a newly registering contact as a location update:

For a received REGISTER:

1. The message must not have more than 1 Contact header while 1 of those Contact headers includes a reg-id parameter. (failure to pass this condition prompts the Oracle CSM to reply to the requester with a 400 Bad Request).
2. The Supported: header contains **outbound** value
3. The Contact header contains a **reg-id** parameter
4. The Contact header contains a **+sip.instance** parameter

After these steps are affirmed, the Oracle CSM determines if it is the First hop. If there is only one Via: header in the REGISTER, the Oracle CSM determines it is the first hop and continues to perform Outbound Registration Binding processing.

If there is more than 1 Via: header in the REGISTER message, the Oracle CSM performs additional validation by checking that a Path: header corresponding to the last Via: includes an ob URI parameter, Outbound Registration Binding may continue.

If the Oracle CSM is neither the first hop nor finds an ob URI in Path headers, it replies to the UA's REGISTER with a 439 First Hop Lack Outbound Support reply.

reREGISTER Example

The user (AoR) bob@example.com registers from a device +sip.instance=<urn:uuid:0001> with a reg-id = "1", contact URI = sip:1.1.1.1:5060. A binding is created for bob@example.com+<urn:uuid:0001>+reg-id=1 at sip:1.1.1.1.:5060.

Next, Bob@example.com sends a reREGISTER with the same instance-id but with a different reg-id = 2 and contact URI = sip:2.2.2.2:5060.

The previous binding is removed. A binding for the new contact URI and reg-id is created. bob@example.com+<urn:uuid:0001>+reg-id=2 at sip:2.2.2.2:5060

Outbound Registration Binding Processing

An outbound registration binding is created between the AoR, instance-id, reg-id, Contact URI, and other contact parameters. This binding also stores the Path: header.

Matching re-registrations update the local registration cache as expected. REGISTER messages are replied to including a Require: header containing the outbound option-tag.

If the Oracle CSM receives requests for the same AOR with some registrations with reg-id + instance-id and some without them, the Oracle CSM will store them both as separate Contacts for the AOR; The AoR+sip.instance+reg-id combination becomes the key to this entry.

ENUM Database Update

When a REGISTER message is received:

1. The ENUM user database is queried for the AoR and any existing entries.
2. If there are any entries with the same instance-id as the current REGISTER request in the ENUM query response, then those entries will be marked for subsequent removal in the ENUM database.
3. The ENUM database is updated with a NAPTR request. This request adds the new Contact URI for that AOR+instance-id and removes any existing entries for the same AOR+instance-id.

NAPTR Update Format

The ENUM database update includes the instance-id and reg-id when those parameters are present in a registration. These values are appended to the regex replacement field. For example:

```
!^.*$!sip:3556-1cdstqjt90hve@172.16.101.62:5060;sip.instance=<urn:uuid:00000000-0000-1000-8000-000A95A0E128>;reg-id=1;ts=1326568408;!
```

ACLI Instructions

ENUM Configuration

First the server used for authentication and as the User Subscriber Database is created.

To configure the ENUM Configuration:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the media-related configurations.


```
ACMEPACKET(configure)# session-router
```
3. Type **enum-config** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# enum-config
ACMEPACKET(enum-config)#
```

You may now begin configuring the enum-config configuration element.
4. **name**—Set a name to use to reference this enum configuration from within the Oracle CSM.
5. **top-level-domain**—Enter the domain which this ENUM server(s) services and returns results for.
6. **realm-id**—Enter the realm name where this ENUM server exists.
7. **enum-servers**—Enter the IP address of one or more ENUM servers used for registration. Multiple entries are separated by commas.
8. **service-type**—Leave this as its default.
9. **ttl**—Leave this at the default of 0 to set the TTL value (in seconds) for NAPTR entries as populated when sending a DNS update to the ENUM server.
10. **order**—Enter the value to populate the order field with when sending NAPTR entries to the ENUM server.
11. **preference**—Enter the value to populate the preference field with when sending NAPTR entries to the ENUM server.
12. Type **done** when finished.

SIP Authentication Profile

To configure the SIP Authentication Profile:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the media-related configurations.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-authentication-profile** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-authentication-profile
ACMEPACKET(sip-authentication-profile)#
```

You may now begin configuring the SIP Authentication Profile configuration element.

4. **name**—Enter the name of this SIP authentication profile that will be referenced from a SIP registrar (or SIP interface).
5. **methods**—Enter all the methods that should be authenticated. Enclose multiple methods in quotes and separated by commas.
6. **anonymous-methods**—Enter the methods from anonymous users that require authentication. Enclose multiple methods in quotes and separated by commas.
7. **digest-realm**—enter the digest realm sent in an authentication challenge (401/407) sent to a UA. This is required in ENUM/DNS deployments.
8. **credential-retrieval-method**—Enter ENUM-TXT.
9. **credential-retrieval-config**—Enter the *enum-config* name used for retrieving authentication data.
10. Type **done** when finished.

SIP Registrar

To configure the Oracle CSM to act as a SIP Registrar:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-registrar** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)#
```
4. **name**—Enter a name for this SIP registrar configuration element.
5. **state**—Set this to **enabled** to use this SIP registrar configuration element.
6. **domains**—Enter one or more domains in the R-URI this configuration element handles. Wildcards are valid for this parameter. Multiple entries can be entered in quotes, separated by commas.
7. **subscriber-database-method**—Set this to **DDNS**.
8. **subscriber-database-config**—Enter the enum-config configuration element *name* that will handle REGISTER messages for this domain. This should be the same element used for requesting authentication data.
9. **authentication-profile**—Enter a *sip-authentication-profile* configuration element's *name*. The *sip authentication profile* object referenced here will be looked up for a REGISTER message with a matching *domain* in the request URI. You may also leave this blank for the receiving SIP Interface to handle which messages require authentication if so configured.
10. **location-update-interval**—Keep or change from the default of 1400 minutes (1 day).
11. Type **done** when finished.

Maximum Number of Contacts

To configure a sip-registrar with a maximum of 10 contacts per AOR:

1. From superuser mode, use the following command sequence to access *sip-registrar* element.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
```

```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)# select
Select the registrar you want to configure.
```

- Specify the number of contacts.

```
AACMEPACKET(sip-registrar)# max-contacts-per-aor 10
AACMEPACKET(sip-registrar)# done
```

To configure local response for the Oracle CSM to issue when max-contacts-per-aor is exceeded:

- From superuser mode, use the following command sequence to access *local-response* and add an entry.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# local-response-map
```

- Access the entries configuration.

```
ACMEPACKET(local-response-map)# entries
```

- Specify the local error you need to configure.

```
ACMEPACKET(local-response-map-entry)# local-error contacts-per-aor-exceed
```

- Specify the sip-reason for this error.

```
ACMEPACKET(local-response-map-entry)# sip-reason forbidden
```

- Specify the error code for this error.

```
ACMEPACKET(local-response-map-entry)# sip-status 403
ACMEPACKET(local-response-map-entry)# done
```

```
local-response-map-entry
    local-error                contacts-per-aor-exceed
    sip-status                  403
    q850-cause                  0
    sip-reason                  forbidden
    q850-reason
    method
    register-response-expires
ACMEPACKET(local-response-map-entry)# exit
```

- In Superuser mode, type **configure terminal** and press <Enter>.

```
ACMEPACKET# configure terminal
```

- Type **session-router** and press <Enter> to access the session router path.

```
ACMEPACKET(configure)# session-router
```

- Type **sip-config** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.

```
ACMEPACKET(session-router)# sip-config
ACMEPACKET(sip-config)#
```

- Type **select** to continue.

```
ACMEPACKET(sip-interface)# select
ACMEPACKET(sip-interface)#
```

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-interface** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-interface
ACMEPACKET(sip-interface)#
```
4. Type **select** and choose the number of the pre-configured sip interface you want to configure.


```
ACMEPACKET(sip-interface)# select
<realm-id>:
1: private 192.168.101.17:5060
2: public 172.16.101.17:5060

selection: 1
```

OAuth 2.0 Support

The Oracle CSM supports Open Authorization (OAuth) in addition to SIP digest authentication for user authorization within ENUM deployments. Both authorization methods can be operational simultaneously, allowing some users to authorize via OAuth and others via SIP digest. Applicable scenarios include authorizing registrations, subscriptions and invites.

OAuth uses HTTP to provide end users with access to services from OAuth 2.0 protected resources using various clients. OAuth also allows users to authorize third-party access to their services using user-agent redirections rather than sharing username password pairs. Any party presenting the proper “bearer token” can be authenticated. The methodology avoids the use of cryptographic keys, and requires protection from token disclosure during transit and storage. OAuth assumes a *secure* exchange of credential validation information between end points, specifically an OAuth client and server, prior to operation.

The Oracle CSM implements OAuth in compliance with RFCs 6749 and 6750. OAuth typically requires deployment-specific compliance beyond RFC compliance. The Oracle CSM allows for typical deployment environments via configuration.

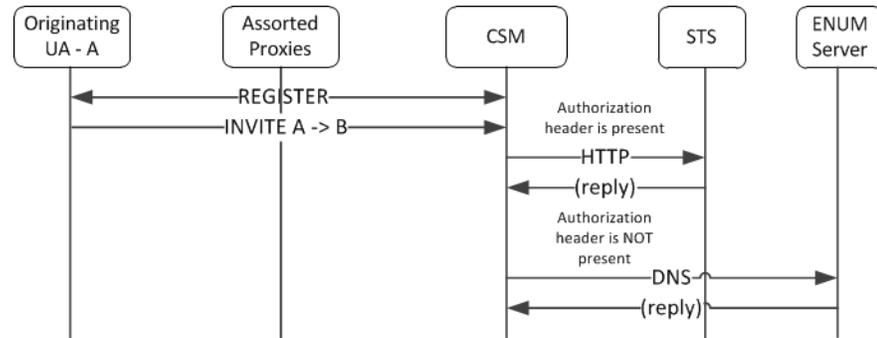
You configure the Oracle CSM to use OAuth by creating OAuth profiles and applying them globally and/or to specific interfaces. Interface profiles take precedence. In addition, you specify the server that the Oracle CSM must contact for authentication using Acme Packet’s SBC Processing Language (SPL). Note that this functionality requires that you upload a script file, provided by Acme Packet, and run an ACLI command to configure the script to use your server. This script file is referred to as a “plug-in”.

OAuth Operation

The Oracle CSM assumes a client requesting OAuth authentication has previously acquired a token from its authorization server. This and token refresh procedures are

performed outside of the scope of Oracle CSM operation. Within the context of OAuth processes, the Oracle CSM performs the functions of a resource server only.

Upon receiving applicable requests from a UA, the Oracle CSM attempts to authenticate using OAuth or Digest. If the request includes a “bearer” string and a token, the Oracle CSM initiates OAuth authentication by sending an HTTP query (GET request) to a Secure Token Service Server (STS), as shown below.



If the STS returns a 200 OK, the Oracle CSM proceeds with authentication. If the STS does not reply or is not reachable, the Oracle CSM replies to the UA with a 500 Internal server error. If the STS replies with any message other than a 200OK, the Oracle CSM replies to the UA with a 403 Forbidden.

The Oracle CSM uses the User ID within the STS 200OK to authenticate and authorize service, as follows:

- For register requests, the system compares this to the user ID field in the “TO” header of the SIP request.
- For other requests, the system compares it to the user ID field in the “FROM” header.

If the user ID matches, the Oracle CSM proceeds with authentication. If this procedure concludes with matches, the Oracle CSM provides service to the UA. If not, it replies to the UA with a 403 forbidden.

Note the following operational caveats:

- If the bearer token is present, but the Oracle CSM is not configured for OAuth, the Oracle CSM responds with a 500 internal service error message. Such misconfiguration includes the OAuth SPL plugin, described below, being absent or disabled.
- If the bearer token is not in the request, the Oracle CSM proceeds with SIP digest authentication.
- If the request arrives over a secure channel, the Oracle CSM follows the procedure defined by the applicable sip-authentication-profile, which may not require authentication.

Configuring OAuth Support

Configuring the Oracle CSM for OAuth support consists of:

- Creating one or more OAuth profiles
- Applying each profile globally or to sip-interfaces

ACLI Instructions

To configure an OAuth profile:

1. From superuser mode, use the following command sequence to access *http-config* element and define your profile.


```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# http-config
ACMEPACKET(http-config)# se1
```
2. Name your profile for reference within your configuration.
3. You can set the **host** parameter to an FQDN or an ip address. If you use an FQDN, the Oracle CSM resolves it to an ip address when the configuration is loaded.
4. The **num-connections** parameter specifies the number of connections to be setup with the STS server. The Oracle CSM establishes this number of connections when it boots or when the configuration is activated.
5. The **max-outstanding-msgs** parameter specifies the maximum number of outstanding messages per connection. An outstanding message is a request from the Oracle CSM that has not had a response. When reaching this threshold, the Oracle CSM stops sending requests on the connection until the number of outstanding messages falls back below this maximum. If the **max-outstanding-msgs** parameter is set to 1, the Oracle CSM waits for a response before sending another request on a connection.
6. The Oracle CSM uses the **port** parameter differently depending on whether the **host** parameter is configured as an IP address or an FQDN. If the **host** parameter is an FQDN name, the Oracle CSM performs a lookup at a DNS server. In this case, you may or may not configure a **port**. If you configure the **port** parameter, the Oracle CSM uses the configured port ignoring any port specified in the record returned by the DNS server. If you do not configure the **port** parameter, the Oracle CSM uses the port returned by the DNS server. If the **host** parameter is set to an ip-address, you must configure the **port** parameter and the Oracle CSM always uses the **port** parameter's value.

Enabling the SPL Plug-in

Enabling the SPL plug-in is a four step process.

1. Upload the SPL plug-in to a Oracle CSM.
2. Add the SPL plug-in to the Oracle CSM configuration.
3. Execute the SPL plug-in on the Oracle CSM.
4. Synchronize the plug-in across HA pairs.

Uploading the Plug-in

The plug-in must be manually FTPed to the Oracle CSM's `/code/sp1` directory using any CLI or GUI-based FTP or SFTP application. The Oracle CSM's FTP/SFTP server may be reached from the system's `wancom` or `eth0` management physical interface.

Adding the Plug-in to Your Configuration

The plug-in must be configured in the *spl-config* configuration element. If multiple plugins are configured on the Oracle CSM, the plug-ins are executed in the order of configuration.

To add the SPL Plugin to the configuration:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **system** and press <Enter> to access the system-level configuration elements.


```
ACMEPACKET(configure)# system
ACMEPACKET(system)#
```
3. Type **spl-config** and press <Enter>.


```
ACMEPACKET(system)# spl-config
ACMEPACKET(spl-config)#
```
4. Type **plugins** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMESYSTEM(spl-config)# plugins
ACMESYSTEM(spl-plugins)#
```
5. Type **name**, a <space>, and the name of the SPL plug-in file.


```
ACMESYSTEM(spl-plugins)#name SMXOAuth2.spl
```
6. Type **done** to save your work.

Executing SPL Files

There are two ways to execute SPL files:

1. Perform a **save-config** and **activate-config** after exiting the configuration menu.
2. 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 planned maintenance windows.

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 ACLI User Interface is booting.

Synchronizing SPL Files Across HA Pairs

When running in an HA configuration, both the active and the standby systems must have the same version of the SPL plugins installed. To facilitate configuring the standby system, you can executing the **synchronize spl** ACLI command (without any arguments) to copy all files in the `/code/spl` directory from the active system to the same directory on the standby, 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 SMXOAuth2.spl
```

The **synchronize spl** command can only be executed from the active system in a HA pair. There is no means to synchronize SPL files automatically during a save and activate of the Oracle CSM.

To synchronize all SPL Plug-ins to the configuration:

1. In Superuser mode, type **synchronize spl** and press <Enter>.


```
ACMEPACKET# synchronize spl
```

Configuring the Plug-in Option

For the SPL to work, you must configure it to recognize your http-server configuration. You do this by setting the `sts-server` option at the sip-registrar. This option is required for plug-in functionality.

To configure the sts-server option:

1. In Superuser mode, type `configure terminal` and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type `session-router` and press <Enter>.


```
ACMEPACKET(configure)# session-router
```
3. Access the target sip-registrar by typing `sip-registrar`, selecting your registrar and pressing <Enter>.


```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)# select
```
4. Type `sts-server=<http-config>`, where "`<http-config>`" is the name of your http configuration element, and press <Enter>.


```
ACMESYSTEM(sip-registrar)# spl-options sts-server=http-server1
```
5. Type `done` to save your work.

Message Routing

The Oracle CSM performs routing in two ways depending on the *routing precedence* parameter in the *sip registrar*. Routing precedence can be set to either **registrar** (ENUM) or **local policy**. Routing precedence is set to `registrar` by default.

Registrar routing uses the configured SIP registrar/ENUM server for destination address queries. Local policy routing lets you configure routing decisions within the Oracle CSM's local policy routing functionality.

If the Oracle CSM is performing any services for the call it performs those services prior to routing.

If, for any reason, the Oracle CSM is unable to proceed with routing a request, it replies to the station that sent the request with a 4xx response.

Registrar Routing

When the *routing precedence* parameter is set to **registrar**, the Oracle CSM is using the ENUM server as a resource within the context of its routing decisions.

When an INVITE arrives, the Oracle CSM issues a query to the ENUM server. If the query's reply includes a match, the Oracle CSM proceeds by forwarding the message via the information in the reply.

Note that you can configure the Oracle CSM to fallback to a local policy lookup if the lookup via the registrar fails. Configure this by adding the **fallback-to-localpolicy** option to the *sip-registrar* configuration element.

For situations where the database routing decision needs to be done in lieu of the default, you can set routing precedence to `local-policy`. Note that you can configure a routing entry that points to an HSS by setting a *policy attribute* with a next-hop of `enum:<server-name>` within the *local-policy*.

Default Egress Realm

The *sip registrar* configuration element should be configured with a default *egress realm id*. This is the name of the *realm config* which defines the IMS control plane through which all Oracle CSMs, ENUM servers, and other network elements communicate and exchange SIP messaging. It is advisable to configure this parameter in order to ensure well defined reachability among Oracle CSMs.

ACLI Instructions

SIP Registrar

To configure a SIP registrar configuration element for message routing:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-registrar** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)#
```
4. Type **select** and choose the number of the pre-configured sip registrar you want to configure.
5. **routing-precedence**— Set this to either **registrar** or **local-policy** depending on your deployment.
6. **egress-realm-id**—Enter the default egress realm for Oracle CSM messaging.
7. Type **done** when finished.

Tel-URI Resolution

The Oracle CSM can initiate number resolution procedures for requests that have tel-URI or SIP-URI (with user=phone) numbers in the R-URI. It does this by querying number resolutions services, including the local routing table(s) or ENUM server(s) to resolve the R-URI to a SIP URI. In addition, the original R-URI may not include a full E.164 number. As such, you can also configure the Oracle CSM to perform a number normalization procedure and ensure it presents a full E.164 number for resolution. Upon successful resolution, the Oracle CSM proceeds with ensuing signaling procedures.

To configure the Oracle CSM to perform these lookups, you create applicable `local-routing-config` or `enum-config` elements and set an option within the `sip-registrar` that specifies a primary and, optionally, a secondary `local-routing-config` or `enum-config` that the `sip-registrar` uses for LRT or ENUM

lookups. If there is no ENUM configuration on the `sip-registrar`, the Oracle CSM forwards applicable requests to a border gateway function via local policy.

Refer to the Net-Net SD ACLI Configuration guide, Session Routing and Load Balancing chapter for complete information on how to configure a `local-routing-config` and an `enum-config`.

Number Lookup Triggers

Use cases that are applicable to number lookups and the associated Oracle CSM procedures include:

- Request from the access side
 1. The Oracle CSM performs originating services
 2. If the R-URI is a tel-URI or SIP-URI (with user=phone) and is not in the Oracle CSM cache, it requests e.164 resolution from the ENUM server(s).
- Request from core side including request for originating services
 1. The Oracle CSM performs originating services
 2. If the R-URI is a tel-URI or SIP-URI (with user=phone) and is not in the Oracle CSM cache, it requests e.164 resolution from the ENUM server(s).
- Request from core side, for terminating services only
 1. If the R-URI is a tel-URI or SIP-URI (with user=phone) and is not in the Oracle CSM cache, it performs a NAPTR lookup.
 2. If the reply indicates the tel-URI or SIP-URI (with user=phone) is not provisioned, the Oracle CSM requests e.164 resolution from the ENUM server(s).

Actions Based on Lookup Results

The Oracle CSM forwards to the resultant SIP-URI under the following conditions:

- The SIP-URI is in the Oracle CSM cache, in which case the Oracle CSM performs terminating services.
- The SIP-URI is not in the Oracle CSM cache, and the Oracle CSM is configured to service the returned domain.
In this case, the Oracle CSM performs the following:
 1. The Oracle CSM issues a NAPTR lookup for the SIP-URI.
 2. The Oracle CSM forwards the message to the correct S-CSCF.
- The SIP-URI is not in the Oracle CSM cache, and the Oracle CSM is not configured to service the returned domain.
In this case, the Oracle CSM performs refers to local policy to forward the message via local policy.

The Oracle CSM may not be able to use the results of the ENUM lookup. Such cases include:

- The lookup at the ENUM servers fail.
- The ENUM response includes the carrier identification code “`cic`” parameter.
- The ENUM response provides a number portability dip indicator “`npdi`” or routing number “`rn`”.

Such cases result in the Oracle CSM sending the request to a gateway, either a BGCF or an MGCF set, based on the rules within the local policy database.

Primary and Secondary ENUM Configs

For the purpose of redundancy, the Oracle CSM allows you to configure these number lookups to use a backup resource in case the lookup at the primary fails. Such scenarios include losing contact with the primary ENUM/LRT server config (query time-out) and the entry is not found at the primary (LRT or ENUM).

ACLI Instructions

To apply primary and secondary number lookup resources to a sip-registrar:

1. From superuser mode, use the following command sequence to access the *sip-registrar* element and select the registrar you want to configure.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)# select
```

2. Specify the resources to use with the options command.

Prepend the option with the “+” character if you have multiple options configured that you want to retain. Running the command without the “+” character causes the system to disable any previously configured options.

To specify primary and secondary ENUM servers:

```
ACMEPACKET(sip-registrar)# options +e164-primary-config=enum:<enum-  
config name>
ACMEPACKET(sip-registrar)# options +e164-secondary-config=enum:<enum-  
config name>
ACMEPACKET(sip-registrar)# done
```

To specify primary and secondary LRT resources:

```
ACMEPACKET(sip-registrar)# options +e164-primary-config=lrt:<lrt-  
config name>
ACMEPACKET(sip-registrar)# options +e164-secondary-config=lrt:<lrt-  
config name>
ACMEPACKET(sip-registrar)# done
```

Bear in mind that an *enum-config* can reference multiple servers. When the Oracle CSM references an *enum-config*, queries follow the normal *enum-config* sequence, checking each referenced server in order. If the lookup is not successful at the primary, the Oracle CSM checks the servers in the registrar’s *e164-secondary-config*.

In addition, each *enum-config* may refer to a different top-level-domain. This allows you to configure the Oracle CSM to successfully perform lookups within two domains.

Extended ENUM Record Length

In some cases, when a user registers a contact from a UA, the Contact: header’s contents are too long to be inserted into the DNS-based user database as presented in the NAPTR record.

To mitigate this, the Oracle CSM stores contact-related metadata, i.e. parameter key and value pairs, in a related TXT record.

If the contents of the Contact: header do not exceed 255 bytes, there is generally no need to extract the metadata to store in an additional TXT record, so the Oracle CSM will not enact this process.

NAPTR and TXT Record Creation and Association

When a user initially registers to the Oracle CSM, the DDNS update sent to the user database will include a NAPTR record and a TXT record. The NAPTR record contains the Contact: header's SIP URI as a regexp replacement. All URI and header parameters from the Contact: header are excluded from the NAPTR record and are inserted into the accompanying TXT record.

The NAPTR record and the TXT record both contain a Oracle CSM-generated key that binds the Contact metadata and the Contact URI sent in the separate records. This common key indicates the data in the NAPTR and TXT record belong to the same registering contact, and is used to recreate the Contact: header for later use. The format of the common key is:

```
p-acme-ckey=<SD-Core-IP>:<integer id>
```

The <SD-Core-IP> is the IP address from which the Oracle CSM communicates with the DNS server. The <integer id> enumerates each contact. Contacts are enumerated in the integer-id element because they can be non-unique when a user behind a NAT registers more than one contact from behind the NAT. For example:

```
TXT "p-acme-ckey=172.16.101.61:1" "$key=value" "^key=value"
NAPTR 1 1 "u" "E2U+sip"!^.*$!sip:642-10u72@172.16.101.61:5060;p-acme-key=172.16.101.61:1!"
```

In the previous example, the key=value pair represent parameters in the Contact header and Contact SIP URI. A \$key= indicates the parameter existed in the Contact: SIP URI. A ^key= indicates the parameter existed in a Contact: header parameter. The Oracle CSM uses this convention to reconstruct the parameters' placement in the original Contact: header.

NAPTR Record Format

Oracle CSMs learn that an associated TXT record exists for the from a NAPTR lookup for the queried SIP URI.

In the DDNS update, the Oracle CSM indicates an associated TXT record by setting the NAPTR resource record with:

- m in the flags field (in addition to the u flag)
- E2U+sip:contact in the service type field

When no additional metadata and no parameters need to stored in ENUM server, the flags field contains u.

TXT Record Retrieval

A Oracle CSM performs a separate query to the ENUM server for the TXT metadata records when it detects the 'm' flag in a NAPTR result (and the one-query-txt-naptr option is not configured). You can set the one-query-txt-naptr option to enabled in the enum-config configuration element to force the Oracle CSM to request the TXT and NAPTR records both in one query from the ENUM server.

Once the Oracle CSM receives the metadata stored in the TXT records, it restores the header and URI parameters that were present on the original registration.

Requirements

BIND 9.8.1-P1 or later hosting the ENUM server supports this feature.

SIP User Parts - RFC 3261 Character Set Support

RFC 3261 specifies the range of characters allowed in a user-part, all of which are supported by the Oracle CSM. ENUM databases, however, do not support all these characters.

By default, the Oracle CSM presents SIP messages to your DNS server unchanged, assuming that your deployment uses telephone numbers only as URI user parts. You configure the Oracle CSM to support the entire range of characters allowed in user parts by enabling it on your ENUM server configuration. Upon configuration, the Oracle CSM grooms the user part appropriately for the ENUM server.

Encoding Alpha-Numerics

If your deployment uses non-numeric characters in user parts, you can explicitly enable the Oracle CSM to change the user part to be compatible with the ENUM database. In these cases, the Oracle CSM encodes SIP message user parts using a proprietary encoding. The Oracle CSM provides the encoded string to the DNS server, which then creates the applicable record(s). The Oracle CSM decodes these strings as necessary during subsequent interactions with the DNS server and the UAs.

In some cases, deployments include SIP messaging in your environment that does not require encoding. These cases include messages with SIP URI user parts composed of:

- tel-uri
- sip-uri with all numeric characters
- sip-uri with all numeric characters except for a leading “+”

In these cases, the Oracle CSM does not encode these user parts. In addition, for all-numeric user parts preceded by the + character, the Oracle CSM strips the + character, reverses the digits, separates the digits with periods, and sends the message to the ENUM server with the user part unencoded.

Multiple DNS Zone Support

For smaller deployments, one can assume a single DNS zone within which all applicable UEs reside. Large deployments, however, may use multiple DNS zones, allowing the network administrator to segregate and more easily manage large numbers of UEs.

When the Oracle CSM encodes the user part, it performs a modulo 1000 operation on a 24-bit checksum of the user part and appends the resulting 3 digits to the encoded user part. You can configure your DNS zones for these digits to organize users into zones.

For example, encoding the user part `acme_user` gives the string `acmeX5Fuser.2.7.7`. Assuming a domain of `acme-ims.com`, the ENUM entry for this user would be

acmex5Fuser.2.7.7.acme-ims.com. Applicable zone configuration can use the .2.7.7 portion of this string to determine this user's zone.

ACLI Instructions

To enable to use of alpha characters in SIP message user parts for a given sip-registrar:

1. From superuser mode, use the following command sequence to access *sip-registrar* element.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# enum-config
ACMEPACKET(enum-config)# select
```

Select the enum-config you want to configure.

2. Specify alpha-numeric name support for this enum-config and specify the number of DNS zones in your deployment.

```
ACMEPACKET(enum-config)# alpha-numeric-user-support enabled
ACMEPACKET(enum-config)# done
```

Note that the parameter `alpha-numeric-user-support` is RTC supported.

Configuring SIP Ping OPTIONS Support

You can configure the Oracle CSM to respond to SIP ping OPTIONS. This support is typically configured on an S-CSCF so it can respond to pings OPTIONS sent by a P-CSCF:

ACLI Instructions

To configure an SIP Options Ping response support:

1. From superuser mode, use the following command sequence to access *ping-response* command on a sip-interface element.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# session-router
ACMEPACKET(session-router)# sip-interface
ACMEPACKET(sip-interface)# sel
```

2. Enable the support with the `ping-response` command.

```
ACMEPACKET(http-config)# ping-response enabled
ACMEPACKET(http-config)# done
```

ping-response—Enable ping-response to allow your device to respond to ping OPTIONS. For example, this feature is useful within hybrid deployment environments on a P-CSCF as a means of verifying the S-CSCF's availability. This configuration allows the S-CSCF to respond to SIP ping OPTIONS.

Local Subscriber Table

A local subscriber table (LST) is an XML formatted file that contains one or more usernames associated with a hash as encrypted or plaintext. The LST is saved locally on the Oracle CSM's file system.

LSTs enable a standalone Oracle CSM node or high-availability (HA) pair to forego relying on an external user database. Thus the Oracle CSM does not need to communicate with a server to authenticate users. This can eliminate the operational complexity of deploying a highly available credential storage system.

LST Runtime Execution

The LST is loaded on boot up when the configuration is appropriately set. Incoming messages thereafter can then be authenticated based on the credentials in the LST. If the Oracle CSM can not load an LST file, three things occur:

1. The following log message is recorded at the NOTICE level:
`LST [table-name] was not loaded - [filename] has error loading XML file`
2. The message stated above is printed on the ACLI.
3. A 503 Response is returned to the UA that sent the initial REGISTER message to the Oracle CSM.

LST Configuration

To configure the Oracle CSM to use LSTs for authentication, you need to create a *local subscriber table* configuration element that identifies that LST. You then need to set the *sip authentication profile* configuration to reference that LST configuration so that when messages requiring authentication are received and processed by a *sip registrar* configuration element, the Oracle CSM will use the identified LST for authentication.

In a *local subscriber table* configuration, you must define an object **name**, identify the specific LST **filename** (and path). If the filename is entered without a path, the Oracle CSM looks in the default LST directory, which is `/code/1st`. If the LST file is located elsewhere on the Oracle CSM, you must specify the filename and absolute path. For example `/code/path/013020121st.xml`.

The corresponding *sip authentication profile* must be set to use the **local subscriber table** configuration element you just created. First set **credential retrieval method** to **local**, set the **digest realm** appropriately (this is required for authentication), and finally set the **credential retrieval config** parameter to the **name** of the *local subscriber table* configuration element that you just created. At this point you may save and activate your configuration.

Unencrypted passwords for each user in the table is computed with the MD5 hash function as follows:

```
MD5(username:digest-realm:password)
```

ACLI Instructions

LST Table

To configure the Oracle CSM to use an LST:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **local-subscriber-table** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# local-subscriber-table
ACMEPACKET(local-subscriber-table)#
```

You may now begin configuring the local subscriber table configuration element.
4. **name**—Enter the name of this local subscriber table configuration element that will be referenced from a SIP registrar configuration element.
5. **filename**—Enter the filename that describes this LST XML file. If no path is given, the Oracle CSM looks in the `/code/lst` directory. You may provide a complete path if the file is located elsewhere.
6. **secret**—Enter the PSK used in encryption and decryption of the passwords in the XML file. Once saved, this value is not echoed back to the screen in plaintext format. See: [LST Subscriber Hash and Encryption \(73\)](#).
7. Type **done** when finished.

SIP authentication profile

To configure the Oracle CSM to utilize an LST, continuing from the previous step:

8. Type **exit** to return to the session router path.
9. Type **sip-authentication-profile** and press <Enter>.
10. Type **select** to choose the existing sip-authentication-profile configuration element you wish to use LST for authentication.


```
ACMESYSTEM(sip-authentication-profile)# select
<name>:
1: name=sipAuthSMX1 digest-realm=acme.com credential-retrieval-
method=loacl

selection: 1
ACMESYSTEM(sip-authentication-profile)#
```

You may now begin configuring the relevant parameters.
11. **digest-realm**—Enter the digest realm used for authenticating here.
12. **credential-retrieval-method**—Set this parameter to **local** to use an LST.
13. **credential-retrieval-config**—Enter the name of the LST configuration you just configured.
14. Type **done** when finished.

LST Redundancy for HA Systems

LSTs must be synchronized between redundant nodes to ensure that the standby node contains identical LST files. You can either SFTP the same LST file to both the

active and standby node, or you can use the synchronize command. The **synchronize** command is always executed from the active system. It copies the specified file from the active to the standby node placing the copy in the same file location on the standby node. Use the **synchronize lst** command as follows:

```
ACMESYSTEM# synchronize lst file.xml
```

Note: The synchronize command does not reload the LST files.

Reloading the LST

After copying a new LST file to the Oracle CSM (and its standby peer), you can reload this newer file from the ACLI using the **refresh lst** command. For example:

```
ACMEPACKET# refresh lst <local-subscriber-table name>
```

Using the **refresh lst** command selects the LST by name to refresh. Alternatively, saving and activating the configuration will reload the configuration as well and should be used when configuration parameters have also changed.

Note: In an HA pair of Oracle CSMs, you must independently execute the **refresh lst** command on both the active and standby systems.

LST File Compression

To save local disk flash space, you can compress the LST XML file using .gz compression. The resultant file must then have an .xml.gz extension.

LST File Format

The LST file format is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<localSubscriberTable encrypt-algo="aes-128-cbc">
<subscriber username="alice@apkt.com" hash="02:5E:78:D8:7E:75:A3:39" encrypted="true"/>
<subscriber username="bob@apkt.com" hash="bc4b2a76b9719d911017c59" encrypted="false"/>
<subscriber username="acme@apkt.com" hash="5d41402abc4b2a76b9719d9" encrypted="false"/>
</localSubscriberTable>
```

The LST file's elements are as follows:

localSubscriberTable

This is the head element in the XML file. Each file can have only one head element. The following attribute is found in this element:

- **encrypt-algo**—This indicates the algorithm type used to encrypt the hash in the XML file. The key for this encryption will be a preshared key and is configurable in the local subscriber table configuration element with the secret parameter.
- The value in this element is for display purposes only.
- Currently AES-128-CBC is the only supported encryption algorithm.

subscriber

This element has the subscriber information. And has the following 3 attributes:

- **username**—The value given in the username attribute must be same as the username that will be sent in the Authorization Header in the Request message from the users. Refer RFC 2617 Http Authentication for details.
- **hash**—The hash provided in the XML must be an MD5 hash of the Username, digest-realm and the password of the user. This is same as the H(A1) described in RFC 2617.

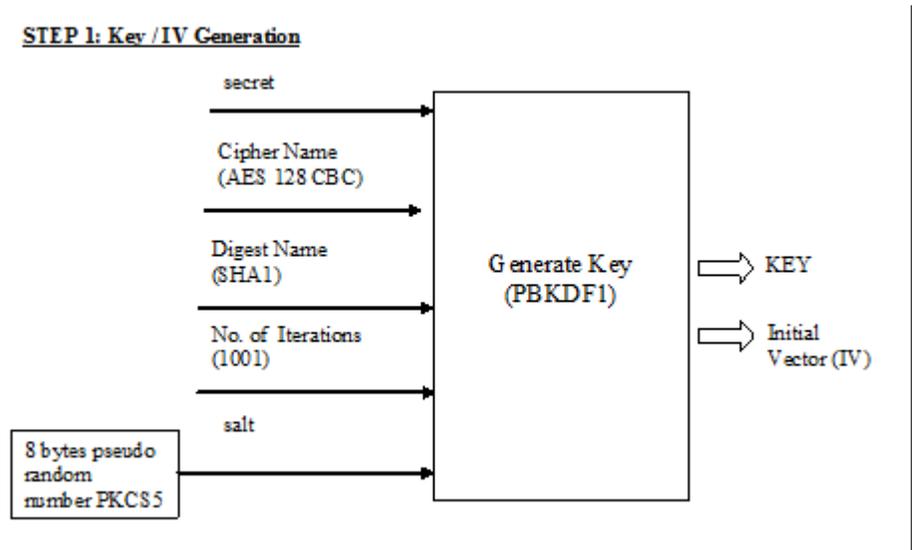
hash = md5(username:digest-realm:password)

- encrypted—The encrypted flag indicates if the "hash" given in the XML file is encrypted or not

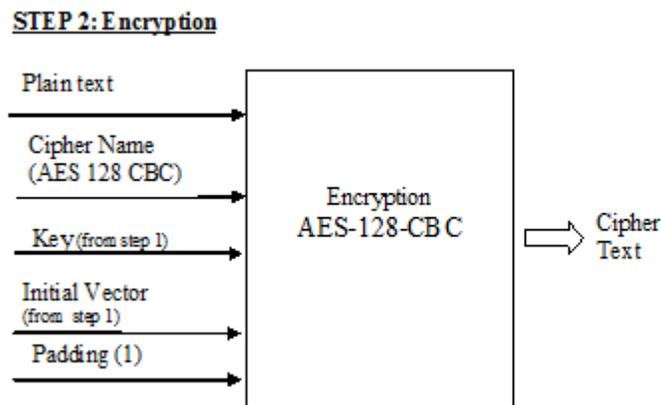
LST Subscriber Hash and Encryption

You may additionally use AES-128 CBC to encrypt the hash in the subscriber element in the LST XML file. The PSK used for encryption is configured in the **secret** parameter and an 8-byte pseudo random number is used as the salt. The LST file must set the encrypted attribute per subscriber element to true. To derive the final encrypted data you place in the XML file, three steps are performed according to the following blocks. The output of the last step, [Formatting final Encrypted Data \(74\)](#), is inserted into the LST files, subscriber element's hash value, when the encrypted attribute is set to true.

Key/Initialization Vector

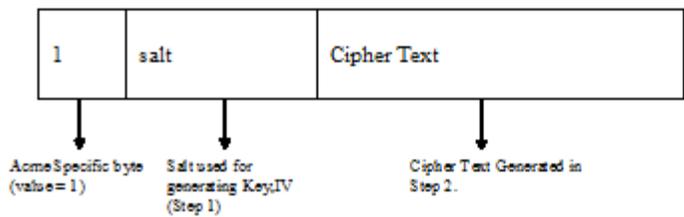


Encryption



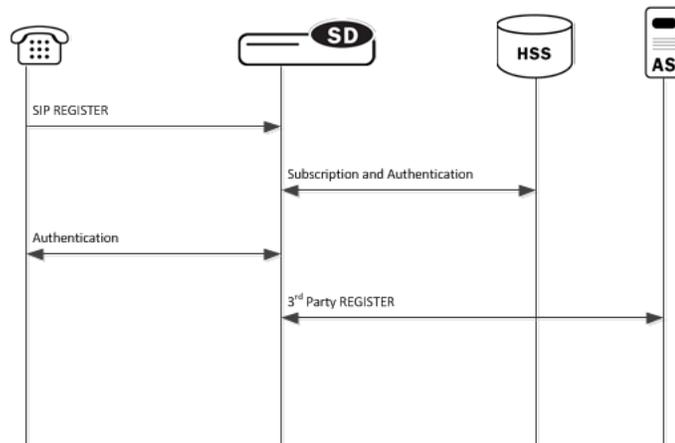
Formatting final Encrypted Data

STEP 3: Final Encrypted Data



Third party registration support on the Oracle CSM provides a mechanism for sending registration information to a third party server. An IM (Instant Messaging) server might be the recipient of a third party REGISTER message.

The Oracle CSM accepts incoming REGISTER requests from UAs. After the UA has been registered with the Oracle CSM, the Oracle CSM sends a third party REGISTER message to a third party server.



The Oracle CSM supports third party registration via two methods:

- For scenarios in which UAs receive iFCs from the HSS and the Oracle CSM's default iFC configuration, the Oracle CSM generates third party registration requests and responses for matching triggers in its iFC evaluation.

Some third party servers may want the UA's entire original request to the Oracle CSM and response from the Oracle CSM to the UA provided to them. The Oracle CSM supports these scenarios, in some cases requiring additional configuration.

- For scenarios in which the UA needs a third party registration that is not explicitly prescribed within iFCs, you can configure a third party server on the Oracle CSM and achieve third party registration support.

For these configurations, the Oracle CSM attempts third party registration to those servers for all UAs that register via the applicable Oracle CSM registrar.

For both methodologies, you must configure all third party servers as session agents.

Third Party Registrations via iFCs

The Oracle CSM performs third party registrations based on the iFC downloaded for the user. If the filter criteria successfully evaluates to a third party server, a third party registration entry is dynamically added in the Oracle CSM. The dynamic entry is automatically deleted if there are no more registrations being handled for that third party registration host.

When third party registration is performed by iFCs, the Oracle CSM generates the registration messages as follows:

- The Contact: header is populated with the URI from the *home server route* configuration of the sip-registrar associated with the registration. If the *home server route* is left blank, the Oracle CSM uses the IP address of the egress interface.
- The From: header of the new REGISTER message is the same as the FROM in the original message.
- The To: header of the new REGISTER message is the the same as the TO in original message (AOR).

Embedded REGISTER

As an option within standard iFC third party registration support, the Oracle CSM supports 3GPP's methodology of embedding the original UE registration (and/or its response from the S-CSCF/Registrar) as a MIME body in the third party REGISTER sent from the S-CSCF to the third party server. This methodology, presented in 3GPP TS 23.218 and 29.228, uses an optional iFC extension ("IncludeRegisterRequest" and "IncludeRegisterResponse") that tells the third party server to expect the entire original REGISTER request and/or REGISTER 200OK in the mime of the third party REGISTER.

Implementation details for this methodology include the following:

- There may be further configuration required on the Oracle CSM.
- The Oracle CSM does not embed original registration requests or responses to any third party server outside its trust domain.
- The HSS or configured iFCs must be preconfigured for embedded third party registrations.

An HSS configuration may not support the optional "IncludeRegisterRequest" and "IncludeRegisterResponse". For these cases, there is a Oracle CSM configuration option that allows you to control this inclusion, as follows:

- If the iFCs specify "inclusion" in an environment where you do not want it, you can set a registrar option to never include the original REGISTER
- If the iFCs do not specify "inclusion" in an environment where you want it, you can set a registrar option to always include the original REGISTER.

You can set these options for either the third party register, the 200 OK, or both.

ACLI Instructions - Third Party Registration via iFCs

Session Agent

To create a session agent to represent the third party server:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **session-agent** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# session-agent
```

```
ACMEPACKET(session-agent)#
```

4. **hostname**—Enter the name for this session agent.
5. **ip-address**—Enter the IP address for this session agent. This value must be the same as the registrar-host parameter in the third party regs configuration element to which this session agent definition corresponds.

Continue configuring this session agent's parameters. Not all session agent functionality is applicable to the Oracle CSM.

6. Type **done** when finished.

SIP Registrar

Option to set the SIP Registrar to perform embedded REGISTRATION support for third party registration:

1. In Superuser mode, type **configure terminal** and press <Enter>.

```
ACMEPACKET# configure terminal
```

2. Type **session-router** and press <Enter> to access the session router path.

```
ACMEPACKET(configure)# session-router
```

3. Type **sip-registrar** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.

```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)#
```

4. Type **select** and choose the number of the pre-configured SIP registrar configuration element you want to configure.

```
ACMEPACKET(sip-registrar)# select
name:
1: registrar1
```

```
selection:1
```

```
ACMEPACKET(sip-registrar)#
```

5. **option +include-register-request**—Set this option to control SIP REGISTER embedding in the third party registration.

```
ACMEPACKET(sip-registrar)#options +include-register-request=true
```

Set this option to true to always embed the original REGISTER in the third party registration.

In some cases, the “include” may already be specified by the iFCs, even though you do not want it used. In these cases, configure the option to false

```
ACMEPACKET(sip-registrar)#options +include-register-request=false
```

6. **option +include-register-response**—Set this option to control SIP REGISTER 200 OK embedding in the third party registration the S-CSCF sends to the AS.

```
ACMEPACKET(sip-registrar)#options +include-register-response=true
```

Set this option to true to always embed the original REGISTER in the third party registration 200 OK.

In some cases, the “include” may already be specified by the iFCs, even though you do not want it used. In these cases, configure the option to false.

```
ACMEPACKET(sip-registrar)#options +include-register-response=false
```

7. Type **done** when finished.

Third Party Registration via ACLI Configuration

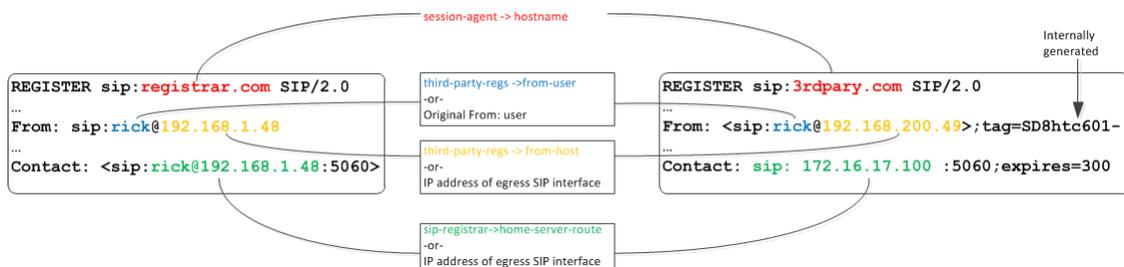
This section specifies the differences between Oracle CSM third party registration support via iFC as opposed to via ACLI configuration.

As is true of the method described above, third party registration is generated by the Oracle CSM on behalf of the user in the To: header of REGISTER request.

When third party registration is generated by ACLI configuration on the Oracle CSM, the registration messages are generated as follows:

- The request URI of the new REGISTER message uses the value of the *hostname* parameter in the *session agent* configuration element.
- The From: header of the new REGISTER message uses the value of the *from-user* parameter in the *third party regs* configuration element as the user portion of the URI. If the *from-user* parameter is left blank, the Oracle CSM uses the user in the original From: header.
- The From: header of the new REGISTER message uses the value of the *from-host* parameter in the *third party regs* configuration element as the host portion of the URI. If the *from-host* parameter is left blank, the Oracle CSM uses the IP address of the egress SIP interface as the host portion of the from header.
- The Contact: header of the new REGISTER message uses the *home server route* parameter in the *sip registrar* configuration element. If the *home server route* parameter is left blank, the Oracle CSM uses the IP address of the egress interface.

See the following diagram:



Third Party Registration Server States

If the third party server does not respond to a REGISTER request, the Oracle CSM adheres to standard SIP session agent retransmission/ timeout procedures. If the third party server is set to out of service, the Oracle CSM attempts connectivity retry procedures. The retry procedures dictate that the Oracle CSM periodically send a REGISTER message to the third party server to check if connectivity has come back. The time interval for checking connectivity to a third party server is set with the *retry interval* parameter. Retries continue forever or until the third party server responds. The retry mechanism may be disabled by setting the *retry interval* parameter to 0.

Note: When using the ACLI generated third party registration method, the time interval for checking connectivity to a third party server is set with the *retry interval* parameter in the *third party regs* configuration element.

When a third party server is out of service, the Oracle CSM maintains a queue of outstanding third party registration requests. When the third party server returns to

service, the Oracle CSM gracefully flushes the queue of outstanding requests. This prevents a registration flood from being directed at the third party server .

Third Party Registration Expiration

The REGISTER message sent from the Oracle CSM to the third party server uses the Expires: value returned from the User Subscriber Database or HSS. The third party server sends a 200 OK message containing Contact bindings and an expires value chosen by the third party server itself. The Oracle CSM checks each contact address to determine if it created it. For those addresses it created (as SD-Contacts), the Expires value from the 200 OK is used as the final value.

Once the expires timer has reached half the expires period as returned from the third party server, the Oracle CSM refreshes the registration.

If the third party server responds to a REGISTER Request with a 423 (Interval Too Brief) response, the Oracle CSM updates the contact's expiration interval to the Min-Expires value of the 423 response. It then submits a new REGISTER Request with the updated expires value.

Defining Third Party Servers

To send third party registrations that are generated via ACLI configuration to a third party server, three configuration elements are required. The primary configuration element is the *third party regs*. One or more may be configured in order to send the REGISTER message to multiple registration servers. You need to configure a *name* and set the *state* to enabled. The *registrar* host must be configured to indicate the value to insert into the Oracle CSM-generated request URI in the REGISTER message.

Note: It is recommended that the list of third party registration servers be restricted to a maximum of 3.

A session agent needs to represent the third party server. Create a session agent as the third party server and note its name. Next, configure the *registrar-host* parameter with a session agent *hostname* in the *third-party-reg* configuration element. This specifies the session agent to be used as the registrar.

Finally, the address of the third party server must be added to the *third-party-registrars* parameter in the *sip-registrar* configuration element. This does not supersede any core Oracle CSM Registrar functionality. It informs the Oracle CSM of the third party server to send messages to after initial registration. Thus the value configured here must exist in the *third-party-regs* configuration element's *registrar-host* parameter list.

ACLI Instructions - Third Party Server Configuration

Recall that the configuration below is only required for scenarios in which the iFC does not explicitly specify registration for the servers you configure below.

Third Party Registrar

To configure a third party server:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.

```
ACMEPACKET(configure)# session-router
```

3. Type **third-party-regs** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# third-party-regs
ACMEPACKET(third-party-regs)#
```
4. **state**—Set this to enabled to use this configuration.
5. **registrar-host**—Set this value to the complementary session agents' *hostname* parameter to include those session agents as third party servers. This parameter may be modified like an options parameter. This value also appears in the request URI of the outgoing REGISTER message being sent to the third party server.
6. **from-user**—Configure this parameter to be the user portion of the From: header of the outgoing REGISTER message being sent to the third party server. Leaving this blank sets the user portion that in the original From: header
7. **from-host**—Configure this parameter to be the host portion of the From: header of the outgoing REGISTER message being sent to the third party server. Leaving this blank sets the host portion to the Oracle CSM's egress SIP interface.
8. **retry-interval**—Enter the number of seconds the Oracle CSM waits before retrying a third party server after a failed registration. Enter **0** to disable this feature.
9. Type **done** when finished.

SIP Registrar

To indicate to a local SIP Registrar when and what third party server to send third party registrations to:

1. In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
2. Type **session-router** and press <Enter> to access the session router path.


```
ACMEPACKET(configure)# session-router
```
3. Type **sip-registrar** and press <Enter>. The system prompt changes to let you know that you can begin configuring individual parameters.


```
ACMEPACKET(session-router)# sip-registrar
ACMEPACKET(sip-registrar)#
```
4. Type **select** and choose the number of the pre-configured SIP registrar configuration element you want to configure.


```
ACMEPACKET(sip-registrar)# select
name:
1: registrar1

selection:1
ACMEPACKET(sip-registrar)#
```
5. **home-server-route**—Enter the value to insert into the REGISTER message's request URI as sent to the third party server. Leaving this blank uses the AoR (or To: header) in the original REGISTER message.
6. **third-party-registrars**—Enter the name of a *third party regs* configuration element *registrar-host* parameter to send third part registrations associated with that SIP registrar.

7. Type **done** when finished.

5 RADIUS Accounting of REGISTERs

CDR Generation for REGISTER Events

The Oracle CSM can generate RADIUS CDRs, per Contact's event, for registration, refresh registration, and registration removal. A single REGISTER message can generate multiple RADIUS CDRs since that message may contain multiple contacts.

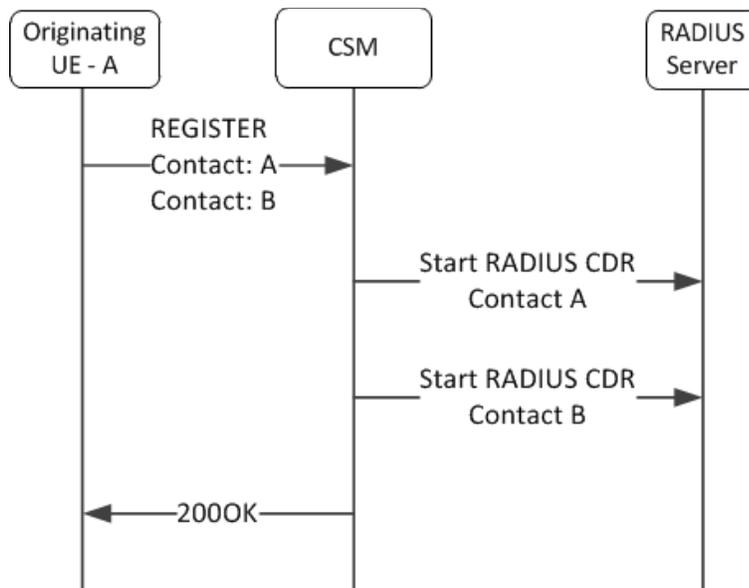
To enable CDR generation for received REGISTERs, you set the **generate event** parameter in the *account-config* configuration element to **register** or **local-register**. The register value may exist with other events such as **invite**.

REGISTER Scenarios

RADIUS CDRs are generated for each registration change per Contact. There are 5 main scenarios which cover CDR creation.

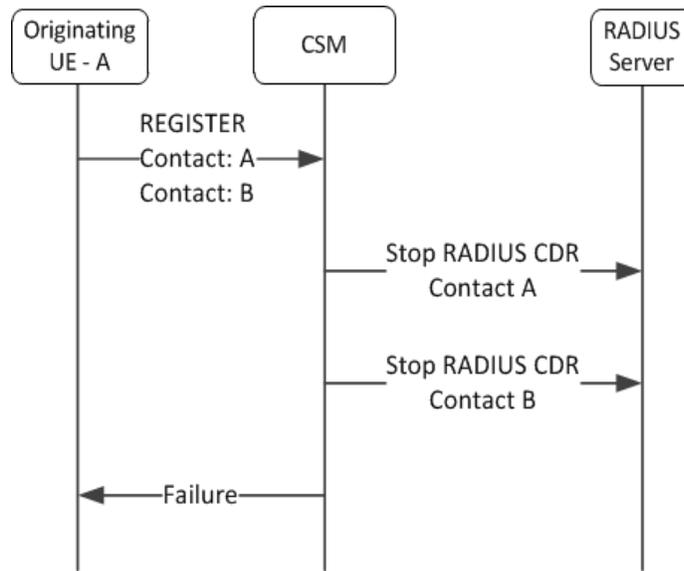
Initial REGISTER

One or more RADIUS Start CDRs are sent to the RADIUS server for each contact in a successful REGISTER message before the Oracle CSM replies to the endpoint with a 200 OK.



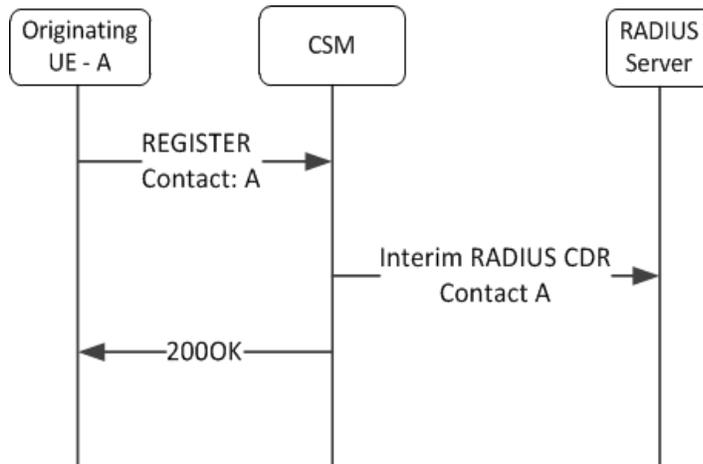
Failed REGISTER

One or more RADIUS Stop CDRs are sent to the RADIUS server for each contact in an unsuccessful REGISTER message before the Oracle CSM replies to the endpoint with a SIP Final Response (4xx or 5xx) message.



REGISTER Refresh

One or more RADIUS Interim CDRs are sent to the RADIUS server for each contact in a successful reREGISTER message before the Oracle CSM replies to the endpoint with a 200 OK. This happens when a database query is made and succeeds.



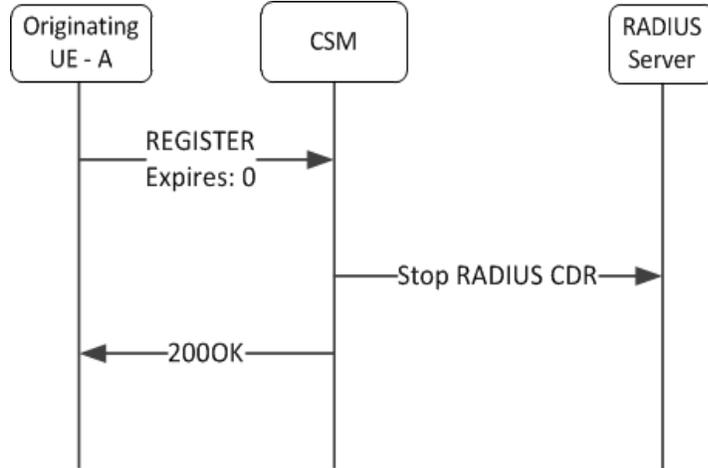
Failed REGISTER Refresh

One or more RADIUS Interim CDRs are sent to the RADIUS server for each contact in an unsuccessful reREGISTER message before the Oracle CSM replies to the endpoint with a SIP Final Response (4xx or 5xx) message.

deREGISTER

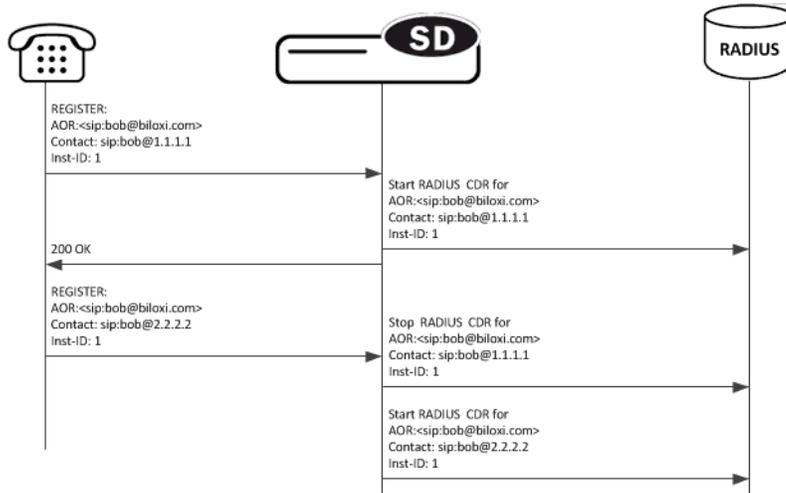
One or more RADIUS Stop CDRs are sent to the RADIUS server for the contacts in a deREGISTER message before the Oracle CSM replies to the endpoint with a 200 OK. The Oracle CSM interprets an expires=0 parameter in a Contact: header as only

removing the registration (and sending a corresponding stop record) for that contact, or an Expires: 0 header prompts Stop RADIUS records for all contacts.



Registration Update

For each Contact’s registration update with an existing Instance-ID and AoR, the Oracle CSM sequentially sends a RADIUS Stop CDR for the original contact address and then a RADIUS Start CDR for the new contact address.

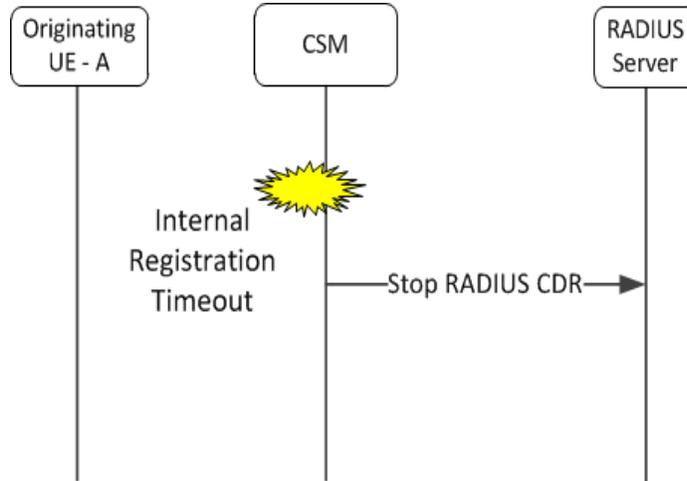


Similarly, when the Oracle CSM receives a REGISTER for an existing Contact and AoR but with a different Instance-ID, the registration is updated by the Oracle CSM. The same corresponding stop and start CDRs are sent to the RADIUS server.

Internal Deregistration

When a condition occurring within the Oracle CSM causes a contact’s registration to be removed, a RADIUS Stop CDR is sent to the RADIUS server. In such a case,

generally no indication is sent to the UA. Examples are if the registration times out, or if the contact is manually removed at the ACLI.



REGISTER VSA Format

The following table lists new VSAs introduced with this feature.

Parameter Name	Dictionary Index	Data Type	Valid Records	Definition
Acme-SIP-Method-Type	235	String	Start, Interim, Stop	This is the SIP method type that is associated to the CDR. Possible values are INVITE, BYE, or REGISTER.
Acme-Domain-Name	236	String	Start, Interim, Stop	This is domain name of the request URI.
Acme-SIP-Contact	237	String	Start, Interim, Stop	This is contact from SIP message. This is not the entire contact header.
Acme-SIP-Expires	238	Integer	Start, Interim	This is the expires value of the Expires: header from the sip message.
Acme-Reason-Phrase	239	String	Interim, Stop	This is the Oracle CSM reason code. This will not be set for all Stop and Interim

The following table lists the new definitions of existing VSAs when CDRs are created on REGISTER messages.

Parameter Name	Dictionary Index	Data Type	Valid Records	Definition
Acme-SIP-Status	71	String	Interim	

Parameter Name	Dictionary Index	Data Type	Valid Records	Definition
Acct-Session-Id	44	String	Start, Interim, Stop	This is the a unique string assigned to the contact. The string is made up of the system name concatenated with a timestamp and an 8 digit hex number. It is of form <system name>-<timestamp>-<number>.
Called-Station-Id		String	Start, Interim, Stop	When in a stop record, this value is populated when an internal reason causes the stop record to be generated. in this case it is the AoR.

The following pairs of Acme-Disconnect-Initiator and Acme-Disconnect-Cause VSAs, as presented in a RADIUS stop CDR that corresponds to a REGISTER message are defined in the Reason column.

Reason	Acme-Disconnect-Initiator	Acme-Disconnect-Cause
UA requested deregistration	1 - CALLING_PARTY_DISCONNECT	1-PW_CAUSE_USER_REQUEST
Contact Times Out	3 - INTERNAL_DISCONNECT	4 -PW_CAUSE_IDLE_TIMEOUT
REGISTER error on establishment.	1 - CALLING_PARTY_DISCONNECT	17-PW - CAUSE_USER_ERROR
RTR from HSS specifies user's private ID	3 - INTERNAL_DISCONNECT	20 - PW_CAUSE_RTR_REQUEST_PRIV
RTR from HSS specifies user's public ID	3 - INTERNAL_DISCONNECT	21 - PW_CAUSE_RTR_REQUEST_PUB
Contact's registration is removed via ACLI command	3 - INTERNAL_DISCONNECT	1-PW_CAUSE_USER_REQUEST
Reuse of ID by a different Contact	3 - INTERNAL_DISCONNECT	22-PW_CAUSE_REUSE_ID

ACLI Instructions

To add CDR generation on REGISTER messages:

- In Superuser mode, type **configure terminal** and press <Enter>.


```
ACMEPACKET# configure terminal
```
- Type **session-router** and press <Enter>.


```
ACMEPACKET(configure)# session-router
```
- Type **account-config** and press <Enter>.


```
ACMEPACKET(session-router)# account-config
```

From this point, you can reach the individual parameters for duplicate RADIUS attribute prevention and for RADIUS attribute selection.
- generate-events**—Set this parameter to register and/or local-register.
- Save and activate your configuration.

Example CDRs

The following examples list when basic registrations create CDRs.

Initial Registration CDR

A Start CDR is created for an initial REGISTER received on the Oracle CSM's 192.168.101.20 interface from 192.168.12.12.

REGISTER message:

```
REGISTER sip:registrar.biloxi.com SIP/2.0
Via: SIP/2.0/UDP bobspc.biloxi.com:5060;branch=z9hG4bknashds7
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Bob <sip:bob@biloxi.com>;tag=456248
Call-ID: 843817637684230@998sdasdh09
User-Agent: Softphone Beta1.5
Authorization: Digest username="Bob", realm=" biloxi.com",
    nonce="84a4cc6f3082121f32b42a2187831a9e",
    response="7587245234b3434cc3412213e5f113a5432"
CSeq: 1826 REGISTER
Contact: <sip:bob@192.168.12.12>
Expires: 7200
Content-Length: 0
```

Start CDR Message, selected attributes:

```
NAS-Identifier = "abc123"
Acct-Status-Type = Start
NAS-IP-Address = 172.16.101.20
NAS-Port = 5060
Acct-Session-Id = "Iapetus-c00000001"
Acme-Session-Ingress-CallId = "843817637684230@998sdasdh09"
Acme-Session-Protocol-Type = "SIP"
Calling-Station-Id = " Bob <sip:bob@biloxi.com>;tag=456248"
Called-Station-Id = " Bob <sip:bob@biloxi.com>"
Acme-Ingress-Network-Interface-Id = "M00"
Acme-Ingress-Vlan-Tag-Value = 0
Acme-Session-Ingress-Realm = "net192"
Acme-Firmware-Version = "SCX6.3.3 F-1 GA (WS Build 18)"
Acme-Local-Time-Zone = "Time Zone Not Set"
Acme-Ingress-Local-Addr = "192.168.101.20:5060"
Acme-Ingress-Remote-Addr = "192.168.12.12:5060"
Acme-SIP-Method-Type = "REGISTER"
Acme-Domain-Name = "registrar.biloxi.com"
Acme-SIP-Contact = "sip:bob@192.168.12.12"
Acme-SIP-Expires = 7200
Acme-CDR-Sequence-Number = 1
Client-IP-Address = 172.30.70.121
Acct-Unique-Session-Id = "51a15d4381d9fe38"
Timestamp = 1329241213
```

Interim Registration CDR

An interim CDR is created for a REGISTER refresh received on the Oracle CSM's 192.168.101.20 interface from 192.168.12.12.

REGISTER message:

```
REGISTER sip:registrar.biloxi.com SIP/2.0
Via: SIP/2.0/UDP bobspc.biloxi.com:5060;branch=z9hG4bknashds7
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Bob <sip:bob@biloxi.com>;tag=456248
```

```

Call-ID: 843817637684230@998sdasdh09
User-Agent: Softphone Beta1.5
Authorization: Digest username="Bob", realm="biloxi.com",
               nonce="84a4cc6f3082121f32b42a2187831a9e",
               response="7587245234b3434cc3412213e5f113a5432"
CSeq: 1826 REGISTER
Contact: <sip:bob@192.168.12.12>
Expires: 7200
Content-Length: 0

```

Interim CDR Message, selected attributes:

```

NAS-Identifier = "abc123"
Acct-Status-Type = Interim-Update
NAS-IP-Address = 172.16.101.20
NAS-Port = 5060
Acct-Session-Id = "Iapetus-c00000001"
Acme-Session-Ingress-CallId = "843817637684230@998sdasdh09"
Acme-Session-Protocol-Type = "SIP"
Calling-Station-Id = " Bob <sip:bob@biloxi.com>;tag=456248"
Called-Station-Id = " Bob <sip:bob@biloxi.com>"
Acme-Ingress-Network-Interface-Id = "M00"
Acme-Ingress-Vlan-Tag-Value = 0
Acme-Session-Ingress-Realm = "net192"
Acme-Firmware-Version = "SCX6.3.3 F-1 GA (WS Build 18)"
Acme-Local-Time-Zone = "Time Zone Not Set"
Acme-Ingress-Local-Addr = "192.168.101.20:5060"
Acme-Ingress-Remote-Addr = "192.168.12.12:5060"
Acme-SIP-Method-Type = "REGISTER"
Acme-Domain-Name = "registrar.biloxi.com"
Acme-SIP-Contact = "sip:bob@192.168.12.12"
Acme-SIP-Expires = 7200
Acme-SIP-Status = "200"
Acme-Reason-Phrase = "OK"
Acme-CDR-Sequence-Number = 1
Client-IP-Address = 172.30.70.121
Acct-Unique-Session-Id = "51a15d4381d9fe38"
Timestamp = 1329241213

```

STOP CDR on REGISTER message

Stop CDRs are generated for REGISTER messages with Expires of 0 from a user agent, a failed initial registration, or Oracle CSM removing the registration. A Stop CDR can take one of two forms:

1. The Stop CDR is generated as part of receiving a response to a REGISTER message.
2. The Stop CDR is generated as part of an internal event such as a Contact timing out.

REGISTER message:

The following REGISTER contains an expires of 0 received on interface 192.168.101.20 from 192.168.12.12:

```

REGISTER sip:registrar.biloxi.com SIP/2.0
Via: SIP/2.0/UDP bobspc.biloxi.com:5060;branch=z9hG4bknashds7
Max-Forwards: 70

```

```

To: Bob <sip:bob@biloxi.com>
From: Bob <sip:bob@biloxi.com>;tag=456248
Call-ID: 843817637684230@998sdasdh09
User-Agent: Softphone Beta1.5
Authorization: Digest username="Bob", realm=" biloxi.com",
               nonce="84a4cc6f3082121f32b42a2187831a9e",
               response="7587245234b3434cc3412213e5f113a5432"
CSeq: 1826 REGISTER
Contact: <sip:bob@192.168.12.12>
Expires: 0
Content-Length: 0

```

Stop CDR Message, selected attributes:

```

NAS-Identifier = "abc123"
Acct-Status-Type = Stop
NAS-IP-Address = 172.16.101.20
NAS-Port = 5060
Acct-Session-Id = " Iapetus-C00000001"
Acme-Session-Ingress-CallId = "843817637684230@998sdasdh09"
Acme-Session-Protocol-Type = "SIP"
Calling-Station-Id = " Bob <sip:bob@biloxi.com>;tag=456248"
Called-Station-Id = " Bob <sip:bob@biloxi.com>"
Acme-Ingress-Network-Interface-Id = "M00"
Acme-Ingress-Vlan-Tag-Value = 0
Acme-Session-Ingress-Realm = "net192"
Acme-Firmware-Version = "SCX6.3.3 F-1 GA (WS Build 18)"
Acme-Local-Time-Zone = "Time Zone Not Set"
Acme-Ingress-Local-Addr = "192.168.101.20:5060"
Acme-Ingress-Remote-Addr = "192.168.12.12:5060"
Acme-Disconnect-Initiator = 1
Acme-Disconnect-Cause = 1
Acme-SIP-Status = "200"
Acme-SIP-Method-Type = "REGISTER"
Acme-Domain-Name = "registrar.biloxi.com"
Acme-SIP-Contact = "sip:bob@192.168.12.12"
Acme-Reason-Phrase = "OK"
Acme-CDR-Sequence-Number = 1
Client-IP-Address = 172.30.70.121
Acct-Unique-Session-Id = "51a15d4381d9fe38"
Timestamp = 1329241213

```

Oracle CSM Initiated Deregistration

Since no SIP Result code is returned to an endpoint when it was internally deregistered by the Oracle CSM, its corresponding VSA is not created and will not appear in the CDR.

No REGISTER message received.

Stop CDR Message, selected attributes:

```

NAS-Identifier = "abc123"
Acct-Status-Type = Stop
NAS-IP-Address = 172.16.101.20
NAS-Port = 5060
Acct-Session-Id = " Iapetus-C00000001"
Called-Station-Id = "sip:bob@biloxi.com"

```

```

Acme-Session-Protocol-Type = "SIP"
Acme-Disconnect-Initiator = 3
Acme-Disconnect-Cause = 4
Acme-SIP-Method-Type = "REGISTER"
Acme-Domain-Name = "registrar.biloxi.com"
Acme-SIP-Contact = "sip:bob@192.168.12.12"
Acme-CDR-Sequence-Number = 1
Client-IP-Address = 172.30.70.121
Acct-Unique-Session-Id = "51a15d4381d9fe38"
Timestamp = 1329241213

```

Local CDR CSV Orientation

This section lists the order of VSAs (and other statistics) in local CDR files.

Start Record

CSV Placement	Attribute Name	ACME VSA ID
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-Forked-Call-Id	171
9	Acme-Session-Generic-Id	40
10	Calling-Station-Id	
11	Called-Station-Id	
12	h323-setup-time	
13	h323-connect-time	
14	Acme-Egress-Network-Interface-Id	139
15	Acme-Egress-Vlan-Tag-Value	140
16	Acme-Ingress-Network-Interface-Id	137
17	Acme-Ingress-Vlan-Tag-Value	138
18	Acme-Session-Egress-Realm	42
19	Acme-Session-Ingress-Realm	41
20	Acme-FlowId_FS1_F	1
21	Acme-FlowType_FS1_F	2
22	Acme-Flow-In-Realm_FS1_F	10
23	Acme-Flow-In-Src-Addr_FS1_F	11
24	Acme-Flow-In-Src-Port_FS1_F	12

CSV Placement	Attribute Name	ACME VSA ID
25	Acme-Flow-In-Dst-Addr_FS1_F	13
26	Acme-Flow-In-Dst-Port_FS1_F	14
27	Acme-Flow-Out-Realm_FS1_F	20
28	Acme-Flow-Out-Src-Addr_FS1_F	21
29	Acme-Flow-Out-Src-Port_FS1_F	22
30	Acme-Flow-Out-Dst-Addr_FS1_F	23
31	Acme-Flow-Out-Dst-Port_FS1_F	24
32	Acme-FlowID_FS1_R	78
33	Acme-FlowType_FS1_R	79
34	Acme-Flow-In-Realm_FS1_R	80
35	Acme-Flow-In-Src-Addr_FS1_R	81
36	Acme-Flow-In-Src-Port_FS1_R	82
37	Acme-Flow-In-Dst-Addr_FS1_R	83
38	Acme-Flow-In-Dst-Port_FS1_R	84
39	Acme-Flow-Out-Realm_FS1_R	85
40	Acme-Flow-Out-Src-Addr_FS1_R	86
41	Acme-Flow-Out-Src-Port_FS1_R	87
42	Acme-Flow-Out-Dst-Addr_FS1_R	88
43	Acme-Flow-Out-Dst-Port_FS1_R	89
44	Acme-FlowID_FS2_F	90
45	Acme-FlowType_FS2_F	91
46	Acme-Flow-In-Realm_FS2_F	92
47	Acme-Flow-In-Src-Addr_FS2_F	93
48	Acme-Flow-In-Src-Port_FS2_F	94
49	Acme-Flow-In-Dst-Addr_FS2_F	95
50	Acme-Flow-In-Dst-Port_FS2_F	96
51	Acme-Flow-Out-Realm_FS2_F	97
52	Acme-Flow-Out-Src-Addr_FS2_F	98
53	Acme-Flow-Out-Src-Port_FS2_F	99
54	Acme-Flow-Out-Dst-Addr_FS2_F	100
55	Acme-Flow-Out-Dst-Port_FS2_F	101
56	Acme-FlowID_FS2_R	112
57	Acme-FlowType_FS2_R	113
58	Acme-Flow-In-Realm_FS2_R	114

CSV Placement	Attribute Name	ACME VSA ID
59	Acme-Flow-In-Src-Addr_FS2_R	115
60	Acme-Flow-In-Src-Port_FS2_R	116
61	Acme-Flow-In-Dst-Addr_FS2_R	117
62	Acme-Flow-In-Dst-Port_FS2_R	118
63	Acme-Flow-Out-Realm_FS2_R	119
64	Acme-Flow-Out-Src-Addr_FS2_R	120
65	Acme-Flow-Out-Src-Port_FS2_R	121
66	Acme-Flow-Out-Dst-Addr_FS2_R	122
67	Acme-Flow-Out-Dst-Port_FS2_R	123
68	Acme-Session-Charging-Vector	54
69	Acme-Session-Charging-Function_Address	55
70	Acme-Firmware-Version	56
71	Acme-Local-Time-Zone	57
72	Acme-Post-Dial-Delay	58
73	Acme-Primary-Routing-Number	64
74	Acme-Originating-Trunk-Group	65
75	Acme-Terminating-Trunk-Group	66
76	Acme-Originating-Trunk-Context	67
77	Acme-Terminating-Trunk-Context	68
78	Acme-P-Asserted-ID	69
79	Acme-Ingress-Local-Addr	74
80	Acme-Ingress-Remote-Addr	75
81	Acme-Egress-Local-Addr	76
82	Acme-Egress-Remote-Addr	77
83	Acme-SIP-Diversion	70
84	Acme-Egress-Final-Routing-Number	134
85	Acme-Session-Ingress-RPH	135
86	Acme-Session-Egress-RPH	136
87	Acme-Custom-VSA-200	200
88	Acme-Custom-VSA-201	201
89	Acme-Custom-VSA-202	202
90	Acme-Custom-VSA-203	203
91	Acme-Custom-VSA-204	204
92	Acme-Custom-VSA-205	205

CSV Placement	Attribute Name	ACME VSA ID
93	Acme-Custom-VSA-206	206
94	Acme-Custom-VSA-207	207
95	Acme-Custom-VSA-208	208
96	Acme-Custom-VSA-209	209
97	Acme-Custom-VSA-210	210
98	Acme-Custom-VSA-211	211
99	Acme-Custom-VSA-212	212
100	Acme-Custom-VSA-213	213
101	Acme-Custom-VSA-214	214
102	Acme-Custom-VSA-215	215
103	Acme-Custom-VSA-216	216
104	Acme-Custom-VSA-217	217
105	Acme-Custom-VSA-218	218
106	Acme-Custom-VSA-219	219
107	Acme-Custom-VSA-220	220
108	Acme-Custom-VSA-221	221
109	Acme-Custom-VSA-222	222
110	Acme-Custom-VSA-223	223
111	Acme-Custom-VSA-224	224
112	Acme-Custom-VSA-225	225
113	Acme-Custom-VSA-226	226
114	Acme-Custom-VSA-227	227
115	Acme-Custom-VSA-228	228
116	Acme-Custom-VSA-229	229
117	Acme-Custom-VSA-230	230
118	Acme-Flow-Calling-Media-Stop-Time_FS1	231
119	Acme-Flow-Called-Media-Stop-Time_FS1	232
120	Acme-Flow-Calling-Media-Stop-Time_FS2	233
121	Acme-Flow-Called-Media-Stop-Time_FS2	234
122	Acme-FlowMediaType_FS1_F	142
123	Acme-FlowMediaType_FS1_R	143
124	Acme-FlowMediaType_FS2_F	144

CSV Placement	Attribute Name	ACME VSA ID
125	Acme-FlowMediaType_FS2_R	145
126	Acme-SIP-Method-Type	235
127	Acme-Domain-Name	236
128	Acme-SIP-Contact	237
129	Acme-SIP-Expires	238
130	Acme-CDR-Sequence-Number	59

Interim Record

CSV Placement	Attribute Name	ACME VSA ID
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
9	Acme-Session-Forked-Call-Id	171
8	Acme-Session-Generic-Id	40
10	Calling-Station-Id	
11	Called-Station-Id	
12	h323-setup-time	
13	h323-connect-time	
14	Acme-Egress-Network-Interface-Id	139
15	Acme-Egress-Vlan-Tag-Value	140
16	Acme-Ingress-Network-Interface-Id	137
17	Acme-Ingress-Vlan-Tag-Value	138
18	Acme-Session-Egress-Realm	42
19	Acme-Session-Ingress-Realm	41
20	Acme-FlowId_FS1_F	1
21	Acme-FlowType_FS1_F	2
22	Acme-Flow-In-Realm_FS1_F	10
23	Acme-Flow-In-Src-Addr_FS1_F	11
24	Acme-Flow-In-Src-Port_FS1_F	12

CSV Placement	Attribute Name	ACME VSA ID
25	Acme-Flow-In-Dst-Addr_FS1_F	13
26	Acme-Flow-In-Dst-Port_FS1_F	14
27	Acme-Flow-Out-Realm_FS1_F	20
28	Acme-Flow-Out-Src-Addr_FS1_F	21
29	Acme-Flow-Out-Src-Port_FS1_F	22
30	Acme-Flow-Out-Dst-Addr_FS1_F	23
31	Acme-Flow-Out-Dst-Port_FS1_F	24
32	Acme-Calling-RTCP-Packets-Lost_FS1	32
33	Acme-Calling-RTCP-Avg-Jitter_FS1	33
34	Acme-Calling-RTCP-Avg-Latency_FS1	34
35	Acme-Calling-RTCP-MaxJitter_FS1	35
36	Acme-Calling-RTCP-MaxLatency_FS1	36
37	Acme-Calling-RTP-Packets-Lost_FS1	37
38	Acme-Calling-RTP-Avg-Jitter_FS1	38
39	Acme-Calling-RTP-MaxJitter_FS1	39
40	Acme-Calling-Octets_FS1	28
41	Acme-Calling-Packets_FS1	29
42	Acme-Calling-R-Factor	151
43	Acme-Calling-MOS	152
44	Acme-FlowID_FS1_R	78
45	Acme-FlowType_FS1_R	79
46	Acme-Flow-In-Realm_FS1_R	80
47	Acme-Flow-In-Src-Addr_FS1_R	81
48	Acme-Flow-In-Src-Port_FS1_R	82
49	Acme-Flow-In-Dst-Addr_FS1_R	83
50	Acme-Flow-In-Dst-Port_FS1_R	84
51	Acme-Flow-Out-Realm_FS1_R	85
52	Acme-Flow-Out-Src-Addr_FS1_R	86
53	Acme-Flow-Out-Src-Port_FS1_R	87
54	Acme-Flow-Out-Dst-Addr_FS1_R	88
55	Acme-Flow-Out-Dst-Port_FS1_R	89
56	Acme-Called-RTCP-Packets-Lost_FS1	46
57	Acme-Called-RTCP-Avg-Jitter_FS1	47
58	Acme-Called-RTCP-Avg-Latency_FS1	48

CSV Placement	Attribute Name	ACME VSA ID
59	Acme-Called-RTCP-MaxJitter_FS1	49
60	Acme-Called-RTCP-MaxLatency_FS1	50
61	Acme-Called-RTP-Packets-Lost_FS1	51
62	Acme-Called-RTP-Avg-Jitter_FS1	52
63	Acme-Called-RTP-MaxJitter_FS1	53
64	Acme-Called-Octets_FS1	44
65	Acme-Called-Packets_FS1	45
66	Acme-Called-R-Factor	153
67	Acme-Called-MOS	154
68	Acme-FlowID_FS2_F	90
69	Acme-FlowType_FS2_F	91
70	Acme-Flow-In-Realm_FS2_F	92
71	Acme-Flow-In-Src-Addr_FS2_F	93
72	Acme-Flow-In-Src-Port_FS2_F	94
73	Acme-Flow-In-Dst-Addr_FS2_F	95
74	Acme-Flow-In-Dst-Port_FS2_F	96
75	Acme-Flow-Out-Realm_FS2_F	97
76	Acme-Flow-Out-Src-Addr_FS2_F	98
77	Acme-Flow-Out-Src-Port_FS2_F	99
78	Acme-Flow-Out-Dst-Addr_FS2_F	100
79	Acme-Flow-Out-Dst-Port_FS2_F	101
80	Acme-Calling-RTCP-Packets-Lost_FS2	104
81	Acme-Calling-RTCP-Avg-Jitter_FS2	105
82	Acme-Calling-RTCP-Avg-Latency_FS2	106
83	Acme-Calling-RTCP-MaxJitter_FS2	107
84	Acme-Calling-RTCP-MaxLatency_FS2	108
85	Acme-Calling-RTP-Packets-Lost_FS2	109
86	Acme-Calling-RTP-Avg-Jitter_FS2	110
87	Acme-Calling-RTP-MaxJitter_FS2	111
88	Acme-Calling-Octets_FS2	102
89	Acme-Calling-Packets_FS2	103
90	Acme-FlowID_FS2_R	112
91	Acme-FlowType_FS2_R	113
92	Acme-Flow-In-Realm_FS2_R	114

CSV Placement	Attribute Name	ACME VSA ID
93	Acme-Flow-In-Src-Addr_FS2_R	115
94	Acme-Flow-In-Src-Port_FS2_R	116
95	Acme-Flow-In-Dst-Addr_FS2_R	117
96	Acme-Flow-In-Dst-Port_FS2_R	118
97	Acme-Flow-Out-Realm_FS2_R	119
98	Acme-Flow-Out-Src-Addr_FS2_R	120
99	Acme-Flow-Out-Src-Port_FS2_R	121
100	Acme-Flow-Out-Dst-Addr_FS2_R	122
101	Acme-Flow-Out-Dst-Port_FS2_R	123
102	Acme-Called-RTCP-Packets-Lost_FS2	126
103	Acme-Called-RTCP-Avg-Jitter_FS2	127
104	Acme-Called-RTCP-Avg-Latency_FS2	128
105	Acme-Called-RTCP-MaxJitter_FS2	129
106	Acme-Called-RTCP-MaxLatency_FS2	130
107	Acme-Called-RTP-Packets-Lost_FS2	131
108	Acme-Called-RTP-Avg-Jitter_FS2	132
109	Acme-Called-RTP-MaxJitter_FS2	133
110	Acme-Called-Octets_FS2	124
111	Acme-Called-Packets_FS2	125
112	Acme-Session-Charging-Vector	54
113	Acme-Session-Charging-Function_Address	55
114	Acme-Firmware-Version	56
115	Acme-Local-Time-Zone	57
116	Acme-Post-Dial-Delay	58
117	Acme-Primary-Routing-Number	64
118	Acme-Originating-Trunk-Group	65
119	Acme-Terminating-Trunk-Group	66
120	Acme-Originating-Trunk-Context	67
121	Acme-Terminating-Trunk-Context	68
122	Acme-P-Asserted-ID	69
123	Acme-Ingress-Local-Addr	74
124	Acme-Ingress-Remote-Addr	75
125	Acme-Egress-Local-Addr	76
126	Acme-Egress-Remote-Addr	77

CSV Placement	Attribute Name	ACME VSA ID
127	Acme-SIP-Diversion	70
128	Acme-Intermediate_Time	63
129	Acct-Session-Time	
130	Acme-Egress-Final-Routing-Number	134
131	Acme-Session-Ingress-RPH	135
132	Acme-Session-Egress-RPH	136
133	Acme-Custom-VSA-200	200
134	Acme-Custom-VSA-201	201
135	Acme-Custom-VSA-202	202
136	Acme-Custom-VSA-203	203
137	Acme-Custom-VSA-204	204
138	Acme-Custom-VSA-205	205
139	Acme-Custom-VSA-206	206
140	Acme-Custom-VSA-207	207
141	Acme-Custom-VSA-208	208
142	Acme-Custom-VSA-209	209
143	Acme-Custom-VSA-210	210
144	Acme-Custom-VSA-211	211
145	Acme-Custom-VSA-212	212
146	Acme-Custom-VSA-213	213
147	Acme-Custom-VSA-214	214
148	Acme-Custom-VSA-215	215
149	Acme-Custom-VSA-216	216
150	Acme-Custom-VSA-217	217
151	Acme-Custom-VSA-218	218
152	Acme-Custom-VSA-219	219
153	Acme-Custom-VSA-220	220
154	Acme-Custom-VSA-221	221
155	Acme-Custom-VSA-222	222
156	Acme-Custom-VSA-223	223
157	Acme-Custom-VSA-224	224
158	Acme-Custom-VSA-225	225
159	Acme-Custom-VSA-226	226
160	Acme-Custom-VSA-227	227

CSV Placement	Attribute Name	ACME VSA ID
161	Acme-Custom-VSA-228	228
162	Acme-Custom-VSA-229	229
163	Acme-Custom-VSA-230	230
164	Acme-Flow-Calling-Media-Stop-Time_FS1	231
165	Acme-Flow-Called-Media-Stop-Time_FS1	232
166	Acme-Flow-Calling-Media-Stop-Time_FS2	233
167	Acme-Flow-Called-Media-Stop-Time_FS2	234
168	Acme-FlowMediaType_FS1_F	142
169	Acme-FlowMediaType_FS1_R	143
170	Acme-FlowMediaType_FS2_F	144
171	Acme-FlowMediaType_FS2_R	145
172	Acme-SIP-Method-Type	235
173	Acme-Domain-Name	236
174	Acme-SIP-Contact	237
175	Acme-SIP-Expires	238
176	Acme-SIP-Status	71
177	Acme-Reason-Phrase	239
178	Acme-CDR-Sequence-Number	59

Stop Record

CSV Placement	Attribute Name	ACME VSA ID
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-Forked-Call-Id	171
9	Acme-Session-Generic-Id	40

CSV Placement	Attribute Name	ACME VSA ID
10	Calling-Station-Id	
11	Called-Station-Id	
12	Acct-Terminate-Cause	
13	Acct-Session-Time	
14	h323-setup-time	
15	h323-connect-time	
16	h323-disconnect-time	
17	h323-disconnect-cause	
18	Acme-Egress-Network-Interface-Id	139
19	Acme-Egress-Vlan-Tag-Value	140
20	Acme-Ingress-Network-Interface-Id	137
21	Acme-Ingress-Vlan-Tag-Value	138
22	Acme-Session-Egress-Realm	42
23	Acme-Session-Ingress-Realm	41
24	Acme-FlowId_FS1_F	1
25	Acme-FlowType_FS1_F	2
26	Acme-Flow-In-Realm_FS1_F	10
27	Acme-Flow-In-Src-Addr_FS1_F	11
28	Acme-Flow-In-Src-Port_FS1_F	12
29	Acme-Flow-In-Dst-Addr_FS1_F	13
30	Acme-Flow-In-Dst-Port_FS1_F	14
31	Acme-Flow-Out-Realm_FS1_F	20
32	Acme-Flow-Out-Src-Addr_FS1_F	21
33	Acme-Flow-Out-Src-Port_FS1_F	22
34	Acme-Flow-Out-Dst-Addr_FS1_F	23
35	Acme-Flow-Out-Dst-Port_FS1_F	24
36	Acme-Calling-RTCP-Packets-Lost_FS1	32
37	Acme-Calling-RTCP-Avg-Jitter_FS1	33
38	Acme-Calling-RTCP-Avg-Latency_FS1	34
39	Acme-Calling-RTCP-MaxJitter_FS1	35
40	Acme-Calling-RTCP-MaxLatency_FS1	36
41	Acme-Calling-RTP-Packets-Lost_FS1	37
42	Acme-Calling-RTP-Avg-Jitter_FS1	38
43	Acme-Calling-RTP-MaxJitter_FS1	39

CSV Placement	Attribute Name	ACME VSA ID
44	Acme-Calling-Octets_FS1	28
45	Acme-Calling-Packets_FS1	29
46	Acme-Calling-R-Factor	151
47	Acme-Calling-MOS	152
48	Acme-FlowID_FS1_R	78
49	Acme-FlowType_FS1_R	79
50	Acme-Flow-In-Realm_FS1_R	80
51	Acme-Flow-In-Src-Addr_FS1_R	81
52	Acme-Flow-In-Src-Port_FS1_R	82
53	Acme-Flow-In-Dst-Addr_FS1_R	83
54	Acme-Flow-In-Dst-Port_FS1_R	84
55	Acme-Flow-Out-Realm_FS1_R	85
56	Acme-Flow-Out-Src-Addr_FS1_R	86
57	Acme-Flow-Out-Src-Port_FS1_R	87
58	Acme-Flow-Out-Dst-Addr_FS1_R	88
59	Acme-Flow-Out-Dst-Port_FS1_R	89
60	Acme-Called-RTCP-Packets-Lost_FS1	46
61	Acme-Called-RTCP-Avg-Jitter_FS1	47
62	Acme-Called-RTCP-Avg-Latency_FS1	48
63	Acme-Called-RTCP-MaxJitter_FS1	49
64	Acme-Called-RTCP-MaxLatency_FS1	50
65	Acme-Called-RTP-Packets-Lost_FS1	51
66	Acme-Called-RTP-Avg-Jitter_FS1	52
67	Acme-Called-RTP-MaxJitter_FS1	53
68	Acme-Called-Octets_FS1	44
69	Acme-Called-Packets_FS1	45
70	Acme-Called-R-Factor	153
71	Acme-Called-MOS	154
72	Acme-FlowID_FS2_F	90
73	Acme-FlowType_FS2_F	91
74	Acme-Flow-In-Realm_FS2_F	92
75	Acme-Flow-In-Src-Addr_FS2_F	93
76	Acme-Flow-In-Src-Port_FS2_F	94
77	Acme-Flow-In-Dst-Addr_FS2_F	95

CSV Placement	Attribute Name	ACME VSA ID
78	Acme-Flow-In-Dst-Port_FS2_F	96
79	Acme-Flow-Out-Realm_FS2_F	97
80	Acme-Flow-Out-Src-Addr_FS2_F	98
81	Acme-Flow-Out-Src-Port_FS2_F	99
82	Acme-Flow-Out-Dst-Addr_FS2_F	100
83	Acme-Flow-Out-Dst-Port_FS2_F	101
84	Acme-Calling-RTCP-Packets-Lost_FS2	104
85	Acme-Calling-RTCP-Avg-Jitter_FS2	105
86	Acme-Calling-RTCP-Avg-Latency_FS2	106
87	Acme-Calling-RTCP-MaxJitter_FS2	107
88	Acme-Calling-RTCP-MaxLatency_FS2	108
89	Acme-Calling-RTP-Packets-Lost_FS2	109
90	Acme-Calling-RTP-Avg-Jitter_FS2	110
91	Acme-Calling-RTP-MaxJitter_FS2	111
92	Acme-Calling-Octets_FS2	102
93	Acme-Calling-Packets_FS2	103
94	Acme-FlowID_FS2_R	112
95	Acme-FlowType_FS2_R	113
96	Acme-Flow-In-Realm_FS2_R	114
97	Acme-Flow-In-Src-Addr_FS2_R	115
98	Acme-Flow-In-Src-Port_FS2_R	116
99	Acme-Flow-In-Dst-Addr_FS2_R	117
100	Acme-Flow-In-Dst-Port_FS2_R	118
101	Acme-Flow-Out-Realm_FS2_R	119
102	Acme-Flow-Out-Src-Addr_FS2_R	120
103	Acme-Flow-Out-Src-Port_FS2_R	121
104	Acme-Flow-Out-Dst-Addr_FS2_R	122
105	Acme-Flow-Out-Dst-Port_FS2_R	123
106	Acme-Called-RTCP-Packets-Lost_FS2	126
107	Acme-Called-RTCP-Avg-Jitter_FS2	127
108	Acme-Called-RTCP-Avg-Latency_FS2	128
109	Acme-Called-RTCP-MaxJitter_FS2	129
110	Acme-Called-RTCP-MaxLatency_FS2	130
111	Acme-Called-RTP-Packets-Lost_FS2	131

CSV Placement	Attribute Name	ACME VSA ID
112	Acme-Called-RTP-Avg-Jitter_FS2	132
113	Acme-Called-RTP-MaxJitter_FS2	133
114	Acme-Called-Octets_FS2	124
115	Acme-Called-Packets_FS2	125
116	Acme-Session-Charging-Vector	54
117	Acme-Session-Charging-Function-Address	55
118	Acme-Firmware-Version	56
119	Acme-Local-Time-Zone	57
120	Acme-Post-Dial-Delay	58
121	Acme-Primary-Routing-Number	64
122	Acme-Originating-Trunk-Group	65
123	Acme-Terminating-Trunk-Group	66
124	Acme-Originating-Trunk-Context	67
125	Acme-Terminating-Trunk-Context	68
126	Acme-P-Asserted-ID	69
127	Acme-Ingress-Local-Addr	74
128	Acme-Ingress-Remote-Addr	75
129	Acme-Egress-Local-Addr	76
130	Acme-Egress-Remote-Addr	77
131	Acme-SIP-Diversion	70
132	Acme-Session-Disposition	60
133	Acme-Disconnect-Initiator	61
134	Acme-Disconnect-Cause	62
135	Acme-SIP-Status	71
136	Acme-Egress-Final-Routing-Number	134
137	Acme-Session-Ingress-RPH	135
138	Acme-Session-Egress-RPH	136
139	Acme-Refer-Call-Transfer-Id	141
140	Acme-Custom-VSA-200	200
141	Acme-Custom-VSA-201	201
142	Acme-Custom-VSA-202	202
143	Acme-Custom-VSA-203	203
144	Acme-Custom-VSA-204	204
145	Acme-Custom-VSA-205	205

CSV Placement	Attribute Name	ACME VSA ID
146	Acme-Custom-VSA-206	206
147	Acme-Custom-VSA-207	207
148	Acme-Custom-VSA-208	208
149	Acme-Custom-VSA-209	209
150	Acme-Custom-VSA-210	210
151	Acme-Custom-VSA-211	211
152	Acme-Custom-VSA-212	212
153	Acme-Custom-VSA-213	213
154	Acme-Custom-VSA-214	214
155	Acme-Custom-VSA-215	215
156	Acme-Custom-VSA-216	216
157	Acme-Custom-VSA-217	217
158	Acme-Custom-VSA-218	218
159	Acme-Custom-VSA-219	219
160	Acme-Custom-VSA-220	220
161	Acme-Custom-VSA-221	221
162	Acme-Custom-VSA-222	222
163	Acme-Custom-VSA-223	223
164	Acme-Custom-VSA-224	224
165	Acme-Custom-VSA-225	225
166	Acme-Custom-VSA-226	226
167	Acme-Custom-VSA-227	227
168	Acme-Custom-VSA-228	228
169	Acme-Custom-VSA-229	229
170	Acme-Custom-VSA-230	230
171	Acme-Flow-Calling-Media-Stop-Time_FS1	231
172	Acme-Flow-Called-Media-Stop-Time_FS1	232
173	Acme-Flow-Calling-Media-Stop-Time_FS2	233
174	Acme-Flow-Called-Media-Stop-Time_FS2	234
175	Acme-FlowMediaType_FS1_F	142
176	Acme-FlowMediaType_FS1_R	143
177	Acme-FlowMediaType_FS2_F	144

CSV Placement	Attribute Name	ACME VSA ID
178	Acme-FlowMediaType_FS2_R	145
179	Acme-SIP-Method-Type	235
180	Acme-Domain-Name	236
181	Acme-SIP-Contact	237
182	Acme-Reason-Phrase	239
183	Acme-CDR-Sequence-Number	59

ACLI Configuration Elements

The following sections describe the Oracle CSM's unique configuration elements.

sip-registrar

Parameters

`name`—Configured name of this sip registrar.

Default: empty

`state`—Running status of this policy-director-group.

Default: enabled

Values: enabled | disabled

`domains`—List of registration domains that this Oracle CSM is responsible for. * means all domains. These domains are compared for an exact match with the domain in the request-uri of the REGISTER message. the wildcard '*' can also be entered as part of this parameter. This is entered as the domains separated by a space in quotes. No quotes required if only one domain is being configured. "+" and "-" are used to add to subtract from the list.

Default: empty

`subscriber-database-method`—Protocol used to connect to User Subscriber Database server.

Default: CX

Values: CX | DDNS | local

`subscriber-database-config`—The configuration element that defines the server used for retrieving user subscriber data. For Cx deployments it is a home-subscriber-server name. For ENUM deployments it is an enum-config name.

Default: empty

`authentication-profile`—Name of the sip-authentication-profile configuration used to retrieve authentication data when an endpoint is not authenticated.

Default: empty

`home-server-route`—The value inserted into the Server Name AVP in an MAR message. This should be entered as a SIP URI as per 3gpp TS 24229 & RFC 3261. The host can be FQDN or IPv4 address, and the port portion should be in the 1025 - 65535 range. Examples: SIP:12.12.12.12:5060

Default: empty

`third-party-registrars`—The third-party-regs configuration element names where third party REGISTER messages will be forwarded to.

Default: empty

`routing-precedence`—Indicates whether INVITE routing lookup should use the user database (via the registrar configuration element) or perform local policy lookup immediately.

Default: registrar

Values: registrar | local-policy

`egress-realm-id`—Indicates the default egress/core realm for SIP messaging.

Default: empty

`location-update-interval`—Sets the maximum period in minutes in which the core-side user subscriber database is refreshed, per user.

Default: 1440

Values: 0-999999999

`ifc-profile`—References the ifc-profile configuration element's name that is applied to this sip-registrar.

`max-contacts-per-aor`—Limit to the number of contacts allowed for a given AOR.

Default: 0 (disabled)

Values: 1 - 256

Path

This sip-registrar configuration element is a element in the session-router path. The full path from the topmost CLI prompt is: `configure terminal > session-router > sip-registrar`.

sip-authentication-profile

Parameters

`name`—Configured name of this sip-authentication profile.

`methods`—List of SIP methods that prompt authentication. This is entered as the methods separated by a space in quotes. No quotes required if only one method is being configured. "+" and "-" are used to add to subtract from the list.

Default: empty

`anonymous-methods`—List of SIP methods that prompt authentication when received from anonymous sources. This is entered as the methods separated by a space in quotes. No quotes required if only one method is being configured. "+" and "-" are used to add or subtract from the list.

Default: empty

`digest-realm`—The value inserted into the digest-realm parameter in an authentication challenge header as sent to UA. (not used for Cx deployments)

Default: empty

`credential-retrieval-method`—Protocol used to connect to the server providing authentication data.

Default: ENUM-TXT

Values: ENUM-TXT | CX

`credential-retrieval-config`—The home-subscriber-server name used for retrieving authentication data.

Default: empty

Path

This sip-authentication-profile configuration element is a element in the session-router path. The full path from the topmost CLI prompt is: `configure terminal > session-router > sip-authentication-profile`.

home-subscriber-server

Parameters

`name`—Configured name of this home subscriber server.

Default: empty

`state`—Running status of this home subscriber server.

Default: enabled

Values: enabled | disabled

`address`—This home subscriber server's IP address.

Default: none

Values: IP address in dotted decimal notation

`port`—This home subscriber server's port.

Default: 80

Values: 1-65535

`realm`—Oracle CSM realm-config name where this home subscriber server exists.

Default: none

`origin-host-identifier`—Used to create segment before the dot in the Origin Host AVP.

Default: none

`origin-realm`—Populates the value of the Origin Realm AVP. Populates the segment after the dot in the Origin Host AVP.

Default: none

`destination-host-identifier`—Used to create segment before the dot in the Destination Host AVP.

Default: none

`watchdog-ka-timer`—

Default: 0

Path

This home-subscriber-server configuration element is a element in the session-router path. The full path from the topmost ACLI prompt is: configure terminal > session-router > home-subscriber-server.

third-party-regs

Parameters

state—Running status of this third party registration configuration element.

Default: enabled

Values: enabled | disabled

name—Configured name of this third party registration configuration element.

Default: none

registrar-host—hostname of the configured session agent that will be third party server. This value is also used in the request-uri that is sent to the third party server.

Default: none

from-user—The user part of the From URI in the REGISTER Request that is sent to the third party server in the REGISTER message. When this parameter is blank the user part of the From header from the incoming REGISTER Request will be used.

Default: none

from-host—The host part of the From URI in the REGISTER Request that is sent the third party server in the REGISTER message. When this parameter is blank the Oracle CSM uses the egress hostname/ IP address as the host.

Default: none

Values: Format this the same as the "registrar-host" in sip-config.

retry-interval—number of seconds the Oracle CSM waits before retrying a 3rd Party Registration server after a failed registration.

Default: 32

Values: 0 - 3600

Path

This third-party-regs configuration element is a element in the session-router path. The full path from the topmost ACLI prompt is: configure terminal > session-router > third-party-regs.

local-subscriber-table

Parameters

name—A given name for this local subscriber table element. This name is referenced from the sip-registrar configuration element when the **credential-retrieval-method** is set to **local**.

filename—The filename of local subscriber table that this element references. If no path is provided, the default location is /code/1st.

secret—PSK used for encrypted passwords. This value is not echoed back to the screen upon viewing the configuration element.

Path The location of this configuration element is: `configure terminal > session-router > local-subscriber-table`.

enum-config

Parameters

`name`—Name for this enum-config to be referenced from within the system.

`top-level-domain`—The domain extension used to query the ENUM servers for this configuration.

`realm-id`—The realm-id is used to determine on which network interface to issue an ENUM query.

`enum-servers`—List of IP address that service the top level domain.

`service-type`—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.

Default: E2U+sip,sip+E2U

`query-method`—the ENUM query distribution strategy

Values: hunt | round-robin

Default: hunt

`timeout`—The total time, in seconds, that should elapse before a query sent to a server (and its retransmissions) will timeout.

Default: 11

`cacheInactivityTimer`—Enter the time interval, in seconds, after which you want cache entries created by ENUM requests deleted, if inactive for this interval.

Default: 3600

Values: 0-999999999

`max-response-size`—The maximum size in bytes for UDP datagram responses

Default: 512

`health-query-number`—The phone number for the ENUM server health query; when this parameter is blank the feature is disabled.

`health-query-interval`—The interval in seconds at which you want to query ENUM server health.

Values: 0-65535

Default: 0

`failover-to`—Name of the enum-config to which you want to failover.

`cache-addl-records`—Set this parameter to **enabled** to add additional records received in an ENUM query to the local DNS cache.

Default: enabled

Values: enabled | disabled

`include-source-info`—Set this parameter to **enabled** to send source URI information to the ENUM server with any ENUM queries.

Default: disabled

Values: enabled | disabled

`t t l`—This value sets the TTL value (in seconds) for NAPTR entries in the local ENUM cache and populates when sending a NAPTR entry to the ENUM server.

Default: 0

Values: 1-2592000

`order`—This parameter value populates the order field with when sending NAPTR entries to the ENUM server.

Default: 1

Values: 0-65535

`preference`—This parameter value populates the preference field with when sending NAPTR entries to the ENUM server.

Default: 1

Values: 0-65535

Path

This enum-config configuration element is a element in the session-router path. The full path from the topmost CLI prompt is: configure terminal > session-router > enum-config.

ifc-profile

Parameters

`name`—A given name for this ifc profile element. This name is referenced from the sip-registrar configuration element's **ifc-support** parameter.

`state`—Running status of this ifc-profile.

Default: enabled

Values: enabled | disabled

`shared-ifc-filename`—The name of the file referenced for shared iFC function.

`default-ifc-filename`—The name of the file referenced for default iFC function. This file may be the same as that used for the shared iFC function.

Path

The location of this configuration element is: configure terminal > session-router > ifc-profile.

regevent-notification-profile

Parameters

`name`—A given name for this registration event notification profile element. This name is referenced from the sip-registrar configuration element.

`min-subscription-duration`—The amount of time, in seconds, before the subscription expires, unless it is refreshed.

Default: 3761 seconds

Values: 180-6000005 seconds

Path

The location of this configuration element is: `configure terminal > session-router > regevent-notification-profile`.

hss-group

Parameters

`name`—Enter the name of the `hss-group` element. This required entry must follow the Name Format, and it must be unique.

`state`—Enable or disable the `hss-group` element.

Default enabled

Values enabled | disabled

`origin-host-identifier`—Set this to a string for use in constructing a unique Origin Host AVP.

`strategy`—Select the HSS allocation options for the `hss-group`. Strategies determine how HSSs will be chosen by this `hss-group` element.

Default hunt

Values

- `hunt`—Selects HSSs in the order in which they are listed. For example, if the first server is online, all traffic is sent to the first server. If the first server is offline, the second server is selected. If the first and second servers are offline, the third server is selected. When the Oracle CSM detects that a higher priority HSS is back in service, it routes all subsequent traffic to that HSS.
- `roundrobin`—Selects each HSS in the order in which they are listed in the dest list, selecting each HSS in turn, one per session. After all HSSs have been used, the first HSS is used again and the cycle continues.
- `failover`—Selects the first sever in the list until failure is detected. Subsequent signaling goes to the next server in the list.

`hss-configs`—Identify the home-subscriber-servers available for use by this `hss-group`. This list can contain as many home subscriber servers as is necessary. An `hss-config` list value must correspond to a valid `hss-group` name in another group or to a valid hostname of a configured home-subscriber-server.

A value you enter here must correspond to a valid group name for a configured home-subscriber-server or a valid hostname or IP address for a configured home-subscriber-server.

hss-group is an element under the `session-router` path. The full path from the topmost CLI prompt is: **configure terminal > session-router > session-group**.

SNMP MIBs and Traps

The following MIBs and traps are supported for the Oracle CSM. Please consult the *Net-Net 4000 S-CX6.3.0 MIB Reference Guide* for more SNMP information.

Acme Packet License MIB (ap-license.mib)

The following table describes the SNMP GET query names for the Acme Packet License MIB (ap-license.mib).

SNMP GET Query Name	Object Identifier Name: Number	Description
Object Identifier Name: apLicenseEntry (1.3.6.1.4.1.9148.3.5.1.1.1)		
apLicenseAuthFeature	apLicenseEntry: 1.3.6.1.4.1.9148.3.5.1.1.1.20	If authorization and authentication is allowed for the Oracle CSM, the value is true. If disabled, the value is false.
apLicenseDatabaseRegFeature	apLicenseEntry: 1.3.6.1.4.1.9148.3.5.1.1.1.21	If the Oracle CSM is configured as a registrar, the value is true. If registrar functionality is not enabled, this value is false.
apLicenseDatabaseRegCap	apLicenseEntry: 1.3.6.1.4.1.9148.3.5.1.1.1.22	The database registration contact capacity.

Acme Packet System Management MIB (ap-smgmt.mib)

The following table describes the SNMP GET query names for the Acme Packet System Management MIB (ap-smgmt.mib).

SNMP GET Query Name	Object Identifier Name: Number	Description
Object Identifier Name: apSysMgmtMIBObjects (1.3.6.1.4.1.9148.3.2.1)		
Object Identifier Name: apSysMgmtGeneralObjects (1.3.6.1.4.1.9148.3.2.1.1)		
apSysSipStatsActiveDatabaseCo ntacts	apSysMgmtGeneralObjects: 1.3.6.1.4.1.9148.3.2.1.1.24.0	Number of database-type contacts in the registration cache.

Enterprise Traps

The following table identifies the proprietary traps that Oracle CSM system supports.

Trap Name: OID	Description
apSysMgmtDatabaseRegCacheCapTrap: 1.3.6.1.4.1.9148.3.2.6.0.76	Generated when the number of database-type contacts stored in the registration cache exceeds the license threshold.
apSysMgmtDatabaseRegCacheCapClearTrap: 1.3.6.1.4.1.9148.3.2.6.0.77	Trap is generated when the number of database-type contacts stored in the registration cache falls below the license threshold.

Oracle CSM Show Commands

show sipd endpoint-ip

The `show sipd endpoint-ip <user | IP address>` command displays information about each endpoint. For a supplied AoR, the Oracle CSM displays all

associated contacts (both access and core side), the expiration of each contact entry and associated 3rd Party Registration information. For example:

```
ACMEPACKET# show sipd endpoint-ip 11111
User <sip:111111@172.16.17.100>
  Contact exp=1198
    UA-Contact: <sip:111111@172.16.17.100:5060> UDP keep-ac1
      realm=net172 local=172.16.101.13:5060 UA=172.16.17.100:5060
    SD-Contact: <sip:111111-s37q249kvluaa@192.168.101.13:5060> realm=net192
    Call-ID: 1-15822@172.16.17.100'
Third Party Registration:
  Third Party Reg User=<sip:111111@172.16.17.100> state: REGISTERED
  Expire Secs=298 seqNum= 1 refreshInterval=300
  Call-ID: d355a67277d9158e7901e46a12719663@192.168.101.13

  Third Party Reg User=<sip:111111@172.16.17.100> state: REGISTERED
  Expire Secs=178 seqNum= 1 refreshInterval=180
  Call-ID: 07ebbdebfd64a48985bb82fa8b4c595@192.168.101.13
```

show sipd third-party

The `show sipd third-party` command displays the current status of third party servers and statistics for messages. The format is:

```
show sipd third-party <all | name>
```

The name argument allows status to be displayed for just the server specified by the name. Not specifying a name results in status being displayed for all third party servers. For example:

```
ACMEPACKET# show sipd third-party-reg all
3rd Party Registrar      SA State  Requests   200OK  Timeouts  Errors
192.168.17.101           INSV      9           9       0         0
192.168.17.102           INSV     14          14       0         0
```

Column definitions are as follows:

- IP Address—IP Address of third party server
- Status—Session Agent State
- Requests—Register requests sent
- 200 OK—200 OK Responses received
- Timeouts—Requests timed out
- Error—Error Responses

show sipd local-subscription

The ACLI `show sipd` command includes an argument that provides information about local subscriptions, as shown below.

```
ACMEPACKET# show sipd local-subscription
19:22:18-152
SIP Local Subscription Status      -- Period -- ----- Lifetime -
-----
                Active  High  Total    Total  PerMax  High
Server Subscription  0     1     1        1      1       1
```

Message Statistics

SUBSCRIBE

Message/Event	Server			Client		
	Recent	Total	PerMax	Recent	Total	PerMax
SUBSCRIBE Requests	2	2	2	0	0	0
Retransmissions	0	0	0	0	0	0
200 OK	1	1	1	0	0	0
403 Forbidden	1	1	1	0	0	0
Response Retrans	0	0	0	0	0	0
Transaction Timeouts	-	-	-	0	0	0
Locally Throttled	-	-	-	0	0	0

Avg Latency=0.000 for 0

Max Latency=0.000

NOTIFY

Message/Event	Server			Client		
	Recent	Total	PerMax	Recent	Total	PerMax
NOTIFY Requests	0	0	0	2	2	2
Retransmissions	0	0	0	10	10	10
200 OK	0	0	0	1	1	1
Transaction Timeouts	-	-	-	0	0	0
Locally Throttled	-	-	-	0	0	0

Avg Latency=0.000 for 0

Max Latency=0.000

You can extend upon this ACLI **show sipd** command to include an argument that provides information about registration event package traffic, as shown below.

ACMEPACKET# show sipd local-subscription regevent

19:23:08-103

SIP Local Subscription Status -- Period -- ----- Lifetime -

	Active	High	Total	Total	PerMax	High
Server Subscription	0	1	1	1	1	1

Message Statistics

SUBSCRIBE

Message/Event	Server			Client		
	Recent	Total	PerMax	Recent	Total	PerMax
SUBSCRIBE Requests	2	2	2	0	0	0
Retransmissions	0	0	0	0	0	0
200 OK	1	1	1	0	0	0
403 Forbidden	1	1	1	0	0	0
Response Retrans	0	0	0	0	0	0
Transaction Timeouts	-	-	-	0	0	0
Locally Throttled	-	-	-	0	0	0

Avg Latency=0.000 for 0

Max Latency=0.000

NOTIFY

----- Server ----- Client -----

Message/Event	Recent	Total	PerMax	Recent	Total	PerMax
	-----	-----	-----	-----	-----	-----
NOTIFY Requests	0	0	0	2	2	2
Retransmissions	0	0	0	10	10	10
200 OK	0	0	0	1	1	1
Transaction Timeouts	-	-	-	0	0	0
Locally Throttled	-	-	-	0	0	0

Avg Latency=0.000 for 0

Max Latency=0.000

The CLI **show registration sipd** command includes an argument that provides information about a specific user's registration(s), as shown below.

```
ACMEPACKET# show registration sipd by-user ra1 detailed
User: sip:ra1@apkt.com
Registered at: 2013-06-05-19:23:40    surrogate User: false
```

Contact Information:

Contact:

```
Name: sip:ra1@apkt.com
Valid: true
Challenged: false
Registered at: 2013-06-05-19:23:40
Last Registered at: 2013-06-05-19:23:40
Expire: 3581
Local expire: 41
Half: 1781
```

```
Registrar IP: 0.0.0.0
Transport: UDP
Secure: false
Local IP: 192.168.101.62:5060
```

User Agent Info:

```
Contact: sip:ra1@192.168.13.1:5060
Realm: net192
IP: 192.168.13.1:5060
```

SD Info:

```
Contact: sip:ra1-1cdstqjt90hve@172.16.101.62:5060
Realm: net172
```

Call-ID: 1-28361@192.168.13.1

Associated URI(s):

```
URI: sip:ra1@apkt.com
Filter Criteria:
Priority: 0
Filter: None specified
Application Server: sip:appserv@apkt.com
```

Reg Event Subscriptions Terminated locally:

Number of Subscriptions: 1

Subscriber: appserv<sip:appserv@apkt.com>;tag=1 state=active
exp=600114

show registration

The show registration command displays cumulative statistics on all current registrations.

```
ACMEPACKET# show registration
15:35:43-177
SIP Registrations          -- Period -- ----- Lifetime -----
                          Active  High  Total      Total  PerMax  High
User Entries               0      0      0          0      0      0
Local Contacts            0      0      0          0      0      0
Via Entries                0      0      0          0      0      0
AURI Entries              0      0      0          0      0      0
Free Map Ports            0      0      0          0      0      0
Used Map Ports            0      0      0          0      0      0
Forwards                  -      -      0          0      0
Refreshes                 -      -      0          0      0
Rejects                   -      -      0          0      0
Timeouts                  -      -      0          0      0
Fwd Postponed             -      -      0          0      0
Fwd Rejected              -      -      0          0      0
Refr Extension            0      0      0          0      0      0
Refresh Extended          -      -      0          0      0
ContactsPerAor Reject-   -      -      0          0      0
Surrogate Regs            0      0      0          0      0      0
Surrogate Sent            -      -      0          0      0
Surrogate Reject          -      -      0          0      0
Surrogate Timeout         -      -      0          0      0
HNT Entries               0      0      0          0      0      0
Non-HNT Entries           0      0      0          0      0      0
Database Regs             0      0      0          0      0      0
DDNS Entries              0      0      0          0      0      0
CX Entries                0      0      0          0      0      0
LocalDB Entries           0      0      0          0      0      0
Unreg Users               0      0      0          0      0      0
```

You can extend upon the show registration command by adding the sipd by-user <username> detail arguments. The resulting output reflects user registration information including downloaded IFCs. For example:

```
ACMEPACKET# show registration sipd by-user +19999092907 d

Registration Cache (Detailed view)    MON JUN 25 2012  13:47:46

User: sip:+19999092907@mobile.com
Registered at: 2012-06-25-13:43:50    surrogate User: false

Contact Information:
Contact:
Name: sip:+19999092907@mobile.com
```

```

Valid: true
Challenged: false
Registered at: 2012-06-25-13:43:50
Last Registered at: 2012-06-25-13:47:30
Expire: 48
Local expire: 13

Registrar IP: 0.0.0.0
Transport: UDP
Secure: false
Local IP: 155.212.214.175:5060

User Agent Info:
Contact:
sip:+19999092907@50.76.51.62:5762;transport=udp;acme_nat=+19999092907
+50.76.51.62@10.1.10.20:5762
  Realm: access
  IP: 50.76.51.62:5762

SD Info:
Contact: sip:+19999092907-rb8tulsvb3u72@108.108.108.108:5060
  Realm: core

Call-ID: H_yvkgTAAA@10.1.10.20

Associated URI(s):
  URI: sip:+19999092907@mobile.com

Filter Criteria:
Priority: 0
  Filter: ((case == 'Originating Registered') and (method == INVITE)
and ('Accept-Contact'=='+g.app2app')) or
          ((case == 'Originating Registered') and (method == INVITE)
and ('Contact'=='+g.app2app')) or
          ((case == 'Originating Registered') and (method == INVITE)
and ('P-Message-Auth'=='.*')) or
          ((case == 'Originating Registered') and (method == INVITE)
and ('P-Application-ID'=='.*'))
  Application Server: sip:pza.mobile.com:5280

Reg Event Subscriptions Received by Registrar:
Number of Subscriptions : 2
Subscriber: sip:appserv@192.168.13.1:5060; state=active; exp=59978
Subscriber: sip:pcscf@192.168.13.1:5060; state=active; exp=978

```

show home-subscriber-server

The show home-subscriber-server command displays cumulative statistics on all currently configured HSS servers.

```
show home-subscriber-server [stats <hss-name>| group group-name ]
```

This command allows you to gather a set of information commonly requested by the Acme Packet TAC when troubleshooting customers.

The show home-subscriber-server command with no arguments displays the status of each HSS as well as the number of transactions and connections per HSS. For example:

```
ACMEPACKET# show home-subscriber-server
Name                Local-Address        Server-Address        Status
hss1                 192.168.207.21:45463 192.168.200.232:3872 Up
-----
```

```
18:53:25-105
HSS Status          -- Period -- ----- Lifetime -----
                   Active  High  Total    Total  PerMax  High
Client Trans        0     1     4       12152   8       1
Server Trans        0     0     0         7     2       1
Connections         1     1     0         53    2       1
```

Note that the Connections statistic indicates the number of connections after successful CER/CEA handshake.

The table below documents the states the Oracle CSM reports on each HSS.

Field	Description
Active	This status is related to HSS failover and load balancing configurations. The diameter connection is up and being used.
Standby	This status is related to HSS failover and load balancing configurations. The diameter connection is up, but is not being used.
Pending	The Oracle CSM has sent a CER and is waiting for a CEA response.
Inactive	The Oracle CSM has sent a CER but has not received a CEA response.
Down	The Oracle CSM is not attempting to establish a connection with the HSS.

The show home-subscriber-server command with the stats argument displays the number of transactions and connections per HSS as well as the number of messages exchanged with all HSS servers per message type. For example:

```
ACMEPACKET# show home-subscriber-server stats

veloster2# show home-subscriber-server stats
Name                Local-Address        Server-Address        Status
hss1                 192.168.207.21:45463 192.168.200.232:3872 Up
-----

18:55:03-103
HSS Status          -- Period -- ----- Lifetime -----
                   Active  High  Total    Total  PerMax  High
Client Trans        1     1     5       12157   8       1
Server Trans        0     0     0         7     2       1
Connections         1     1     0         53    2       1

                                ---- Lifetime ----
                                Recent    Total    PerMax
UAR                                0         3         1
  SUBSEQ_REG (2002)                0         3         1
SAR                                0         6         3
```

SUCCESS (2001)	0	6	3
MAR	0	4	2
SUCCESS (2001)	0	4	2
LIR	0	1	1
SUCCESS (2001)	0	1	1
RTR	0	1	1
SUCCESS (2001)	0	1	1
PPR	0	1	1
SUCCESS (2001)	0	1	1
CER	0	55	3
SUCCESS (2001)	0	53	2
DWR	5	12088	5
SUCCESS (2001)	4	12041	5
ERR_TIMEOUT	0	46	1
DWR Recv	0	5	2
SUCCESS (2001)	0	5	2
TCP Failures	0	267	6

By entering the name of a specific HSS as an argument, the ACLI displays all HSS data for that server only. For example:

```
ACMESYSTEM# show home-subscriber-server stats hss1
```

The show home-subscriber-server command with the group argument displays the number of transactions and connections per the HSS group you specify in the command. For example:

```
ACMEPACKET# show home-subscriber-server group hss-group1
display grp hss-group1
HSS 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
UAR                 0     0     0
SAR                 0     0     0
MAR                 0     0     0
LIR                 0     0     0
RTR                 0     0     0
PPR                 0     0     0
Sent Requests      0     0     0
Sent Req Accepted  0     0     0
Sent Req Rejected  0     0     0
Sent Req Expired   0     0     0
Sent Req Error     0     0     0
Recv Requests     0     0     0
Recv Req Accepted  0     0     0
Recv Req Rejected  0     0     0
HSS Errors        0     0     0
```

show http-server

The ACLI **show http-server** command provides basic OAuth information as shown below. The command without arguments displays basis statistics on all servers.

```
ACMEPACKET# show http-server
```

Name	Server-Address	Status
sk	host.httpsrv.com	Up
sk1	192.168.19.1:8886	Up
sk2	192.168.19.1:8887	Up
sk3	192.168.19.1:8889	Up

```
12:56:41-184
```

```
HTTP 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

You can extend upon this command to get detailed global statistics by adding the **stats** argument to the end of this command.

```
ACMEPACKET# show http-server stats
```

Name	Server-Address	Status
sk	host.httpsrv.com	Up
sk1	192.168.19.1:8886	Up
sk2	192.168.19.1:8887	Up
sk3	192.168.19.1:8889	Up

```
HTTP 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	1	1	1	1	1	1
Connections	1	1	1	1	1	1

```
----- Lifetime -----
```

	Recent	Total	PerMax
Sent Requests	0	0	0
Sent Req Accepted	0	0	0
Sent Req Rejected	0	0	0
Sent Req Expired	0	0	0
HTTP Errors	0	0	0

You can limit this output to a single server by appending the command with the name of that server.

```
ACMEPACKET# show http-server stats http-server1
```

```
Name = http-server1
```

```
-----
```

Server-Address	Status
192.168.19.1:8886	Up

```
-----
```

```
12:56:41-184
```

HTTP 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			
Sent Requests	0	0	0			
Sent Req Accepted	0	0	0			
Sent Req Rejected	0	0	0			
Sent Req Expired	0	0	0			
HTTP Errors	0	0	0			

Supporting Configuration

The following configuration elements which are not mentioned in this guide are required for the Oracle CSM to function. Please refer to the Net-Net 4000 ACLI Configuration Guide for details about configuring all supporting elements.

- network-interface
- physical-interface
- realm-config
- sip-config
- system-config

The following configuration elements are mentioned in this guide briefly and still require configuration:

- local-policy
- session-agent
- sip-interface

Verify Config

The Oracle CSM performs application specific verification checks when you save a config with the save-config ACLI command. These checks are in addition to baseline Net-Net SBC verification checks.

sip authentication profile (CX)

If "session-router > sip-authentication-profile > credential-retrieval-method" = CX then confirm

"session-router > sip-authentication-profile > credential-retrieval-config" value = any existing "session-router > home-subscriber-server configuration > name" value

Error

If the above check fails:

1. A WARNING is displayed on the ACLI.
2. An INFO log message is generated.

sip authentication profile (ENUM)

If “session-router > sip-authentication-profile > credential-retrieval-method” = ENUM-TXT then confirm

“session-router > sip-authentication-profile > credential-retrieval-config” value = any existing “session-router > enum-config > name” value

Error

If the above check fails:

1. A WARNING is displayed on the ACLI.
2. An INFO log message is generated.

sip authentication profile (Local)

If “session-router > sip-authentication-profile > credential-retrieval-method” = local then confirm

“session-router > sip-authentication-profile > credential-retrieval-config” =

“session-router > local-subscriber-table > ame” Error

If the above check fails:

1. A WARNING is displayed on the ACLI.
2. An INFO log message is generated.

sip-registrar

If “session-router > sip-registrar > subscriber-database-method” = DDNS then confirm

“session-router > sip-registrar > subscriber-database-config” value = any existing “session-router > enum-config > name” value

Error

If the above check fails:

1. A WARNING is displayed on the ACLI.
2. An INFO log message is generated.

sip-registrar

If “session-router > sip-registrar > authentication-profile” is configured, then confirm its value is any existing:

“session-router > sip-authentication-profile > name” value

Error

If the above check fails:

1. A WARNING is displayed on the ACLI.
2. An INFO log message is generated.

Memory Utilization

The Net-Net CSM limits resource utilization to maintain operational stability, as follows:

When heap utilization exceeds 75%, the Net-Net CSM no longer accepts new registrations. The Net-Net CSM replies to these messages with 5xx messages.

Note that the Net-Net CSM continues to accept registration refreshes, new calls and new subscriptions.

- When heap utilization exceeds 85%, the Net-Net CSM no longer accepts new calls and subscriptions. The Net-Net CSM replies to these messages with 5xx messages.

Note that the Net-Net CSM continues to accept registration refreshes. In addition, the Net-Net CSM continues to accept messages within existing dialogs.

The user can determine current memory utilization using the following command and referring to the `heap utilization` value, towards the bottom of the command's output.

```
ACMEPACKET# show platform heap-statistics
```

The user can disable this functionality using the option shown below.

```
ACMEPACKET(sip-config)# +options disable-memory-overload-protect
```

