

Oracle® Communications Session Border Controller Call Traffic Monitoring Guide



Release S-CZ7.3.0
January 2018



Copyright © 2014, 2018, 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, then 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 about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface: About this Guide

1	Call Traffic Monitoring	
	SelectiveCall Recording SIPREC	1-1
	SIPREC Feature	1-1
	Configuring SIPREC	1-2
	Session Recording Server (SRS)	1-2
	Session Recording Group	1-3
	Load Balancing	1-3
	Session Recording Group within Logical Remote Entities	1-4
	Selective Recording	1-4
	High Availability (HA) Support	1-5
	SIPREC Configuration Procedure	1-5
	Session-recording-server Attribute	1-5
	Session-recording-group Attribute (for HA only)	1-7
	Realm-config Attribute	1-9
	Session-agent Attribute	1-10
	Sip-interface Attribute	1-11
	P-Asserted Identity and Diversion Headers in SIPREC Metadata	1-12
	Metadata Contents	1-13
	Show Commands for Recording Sessions	1-14
	Show rec	1-14
	Show rec redundancy	1-15
	Inserting SIP Headers into SIPREC Metadata	1-16
	Sample Metadata	1-16
	Configure SIP Headers for SIPREC Metadata	1-18
	SIPREC Recording Session Refresh	1-18
	Timer_B	1-19
	OPTIONS Request/Response	1-19
	Recording Session Refresh Configuration	1-19
	Codec Negotiation	1-20
	SIPREC Call Flows	1-20

Selective Recording	1-21
Normal Call (recording required)	1-21
Sample SDP and Metadata	1-22
Normal Call (recording not required)	1-23
Early Media Call (recording not required)	1-24
REFER Pass-Through Call (REFER handled by User Agent)	1-26
REFER Call (REFER handled by Oracle Communications Session Border Controller)	1-27
SRS Indicates Busy in Call (recording not required)	1-29
Call Transfer Scenario (recording required)	1-31
Oracle Communications Session Monitor Mediation Engine	1-32
IPFIX	1-33
Communications Monitor Configuration	1-33
Communication Monitor	1-33
TSCF Rekey Profile Configuration	1-35
TLS Profile Configuration	1-36
Oracle Communications Session Monitor Statistics	1-37
Packet Trace	1-39
Packet Trace Remote	1-39
Packet Trace Local	1-40
Packet Trace Scenarios	1-41
Packet Trace for One Endpoint	1-41
Packet Trace for Both Call Legs	1-42
Packet Trace for a Oracle Communications Session Border Controller Signaling Address	1-42
Running Packet Trace	1-43
Configuring a Trace Server	1-44
Starting a Remote Packet Trace	1-44
Stopping a Remote Packet Trace	1-45
Starting a Local Packet Trace	1-45
Stopping a Local Packet Trace	1-46

2 Web-based GUI

Introduction	2-1
Web Browser Support	2-1
Overview	2-1
Logging in	2-1
About Information	2-3
Customizing the Page Display	2-4
Changing Number of Data Items on the Page	2-5
Navigating Pages	2-6

Refresh	2-6
Pop-up Context Menu	2-7
Help Menu	2-7
Tool-Tip Help	2-9
Search for a Record	2-9
Exporting Information to a Text File	2-14
Session Reports	2-15
Session Report Configuration	2-17
Ladder Diagram	2-17
Session Summary	2-19
SIP Message Details	2-20
QoS Statistics	2-21
Registration Reports	2-22
Subscription Reports	2-24
Notable Event Reports	2-26

Preface: About this Guide

The Oracle Communications Session Border Controller Call Traffic Monitoring Guide provides information about monitoring the call traffic on your system.

Related Documentation

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

Document Name	Document Description
Acme Packet 4500 Hardware Installation Guide	Contains information about the components and installation of the Acme Packet 4500.
Acme Packet 3820 Hardware Installation Guide	Contains information about the components and installation of the Acme Packet 3820.
Acme Packet 4600 Hardware Installation Guide	Contains information about the components and installation of the Acme Packet 4600.
Acme Packet 6100 Hardware Installation Guide	Contains information about the components and installation of the Acme Packet 6100.
Acme Packet 6300 Hardware Installation Guide	Contains information about the components and installation of the Acme Packet 6300.
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 of the Service Provider Oracle Communications Session Border Controller.
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 Oracle Communications Session Border Controller 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), Oracle Communication's 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 Oracle Communications Session Border Controller's accounting support, including details about RADIUS and Diameter accounting.
HDR Resource Guide	Contains information about the Oracle Communications Session Border Controller's Historical Data Recording (HDR) feature. This guide includes HDR configuration and system-wide statistical information.

Document Name	Document Description
Administrative Security Essentials	Contains information about the Oracle Communications Session Border Controller's support for its Administrative Security license.
Security Guide	Contains information about security considerations and best practices from a network and application security perspective for the Oracle Communications Session Border Controller family of products.
Installation and Platform Preparation Guide	Contains information about upgrading system images and any pre-boot system provisioning.
Call Traffic Monitoring Guide	Contains information about traffic monitoring and packet traces as collected on the system. This guide also includes WebGUI configuration used for the SIP Monitor and Trace application.

Revision History

Date	Description
October 2015	<ul style="list-style-type: none"> Initial Release Remove MGCP support in documentation
May 2016	<ul style="list-style-type: none"> Adds requirement that any media interface configured with comm-monitor must also belong to a realm.
September 2016	<ul style="list-style-type: none"> Removes Single SRS capacity section
February 2017	<ul style="list-style-type: none"> Updates Egress RTP packet capture information under Packet Trace
April 2017	<ul style="list-style-type: none"> Clarifies that local packet capture does not include RTP
January 2018	<ul style="list-style-type: none"> Removes deprecated Web Server Configuration chapter.

1

Call Traffic Monitoring

SelectiveCall Recording SIPREC

The SIPREC protocol is the protocol used to interact between a Session Recording Client (SRC) (the role performed by Oracle Communications Session Border Controller) and a Session Recording Server (SRS) (a 3rd party call recorder or Oracle Communications Interactive Session Recorder's Record and Store Server (RSS)). It controls the recording of media transmitted in the context of a communications session (CS) between multiple user agents.

SIPREC provides a selective-based call recording solution that increases media and signaling performance on 3rd party call recording servers, more robust failovers, and the ability to selectively record.

Note:

SIPREC isolates the 3rd party recorders from the communication session. The 3rd party recorders can determine whether or not recording is desired.

Note:

The SRC starts a recording session for every call within a configured realm. All call filtering, if desired, must be accomplished by the SRS. The SRS performs the filtering and selection of which sessions it should record.

SIPREC Feature

The SIPREC feature supports active recording, where the Oracle Communications Session Border Controller acting as the SRC, purposefully streams media to the Oracle Communications Interactive Session Recorder's RSS (or 3rd party call recorder) acting as the SRS. The SRC and SRS act as SIP User Agents (UAs). The SRC provides additional information to the SRS to describe the communication sessions, participants and media streams for the recording session to facilitate archival and retrieval of the recorded information.

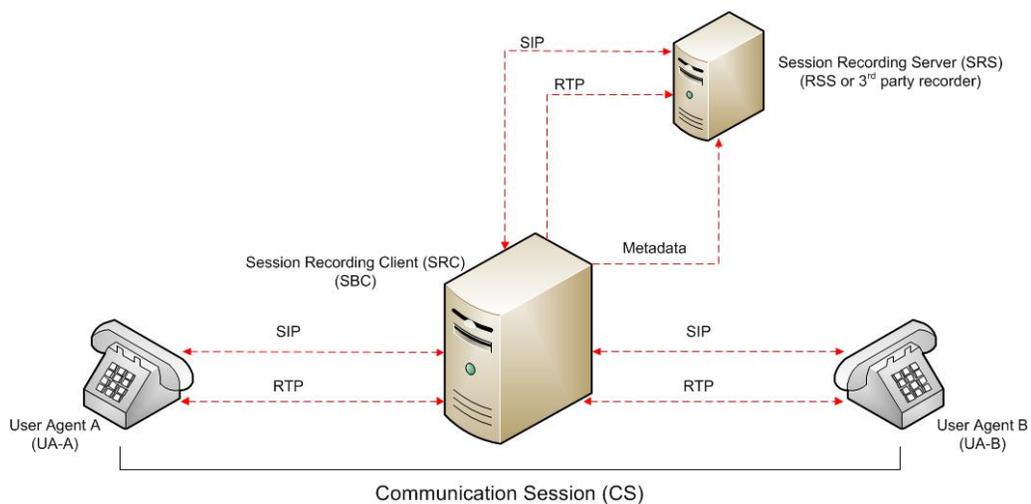
The Oracle Communications Session Border Controller acting as the SRC, is the source for the recorded media. The Oracle Communications Session Border Controller consumes configuration information describing the ecosystem within which it operates. The interface, realm and session agent configuration objects specify the SIPREC configuration. A SIP UA can elect to allow or disallow any network element from recording its media.

During the establishment of a SIP Session, the Oracle Communications Session Border Controller determines if SIPREC is configured for recording the call. If so, it then duplicates the media prior to initiating the session with the SRS. (Media replication is set up prior to the

recording session). The SRS may choose to record, not record, or cancel the recording session, and then communicates via SIP signaling to the Oracle Communications Session Border Controller. If the call is not to be recorded, the SRS signals termination of the recording session.

The Oracle Communications Session Border Controller maintains SIPREC metadata information associated with recording sessions. The recording session metadata describes the current state of the recording session and its communication session(s). It is updated when a change of state in the communication session(s) is observed by the Oracle Communications Session Border Controller. The SRS is responsible for maintaining call history, etc. The Oracle Communications Session Border Controller creates and logs call detail records (CDRs) in the current manner, the 3rd party SRS vendor may collate this information if desired. (For more information about the contents of metadata, see [Metadata Contents](#)).

The following illustration shows two endpoints, User Agent A (UA-A) and User Agent B (UA-B). Their session is being recorded by an SRC (the Oracle Communications Session Border Controller) and an SRS.



Configuring SIPREC

This section defines the information required to configure SIPREC on the Oracle Communications Session Border Controller. It also provides a sample procedure for configuring SIPREC using the Acme Packet Command Line Interface (ACLI).

Session Recording Server (SRS)

The Oracle Communications Interactive Session Recorder's RSS acts as the SRS in the network. A **session-recording-server** attribute under the **session-router** object in the Oracle Communications Session Border Controller ACLI allows you to enable/disable the SRS. This object is the session recording server that receives replicated media and records signaling. Additional parameters for SRS are configured under the **session-agent**, **realm-config**, and **sip-interface** objects. The rules of precedence for which the Oracle Communications Session Border Controller uses these parameters are:

session-agent takes precedence over the **realm-config**, and **realm-config** takes precedence over **sip-interface**.

Each SRS is associated with a **realm-config**. The realm specifies the source interface from which replicated traffic originates. The destination is an IP Port parameter (IP address or hostname with an optional port) that defines the SIP address (request URI) of the actual SRS.

For an additional level of security, Oracle recommends the SRS be configured in its own realm so as to apply a set of access control lists (ACLs) and security for the replicated communication.

Although the Oracle Communications Session Border Controller supports large UDP packets, Oracle recommends the **sip-interface** associated with the SRS realm, be provisioned with a TCP port.

Session Recording Group

The Oracle Communications Session Border Controller uses the **session-recording-group** attribute under the **session-router** object in the CLI to set high availability (HA) for 3rd party call recorders. Using this object, you can define a collection of one or more SRSs. The Oracle Communications Session Border Controller utilizes SIP's transport mechanism and keeps track of statistics on each SRS to manage the distribution of traffic and load balancing. (For more information on Oracle Communications Session Border Controller load balancing in session recording groups, see [Load Balancing](#)). When multiple SRSs are in a session recording group, the Oracle Communications Session Border Controller uses heuristics to intelligently route the recording dialog to one or more SRSs utilizing the selection strategy.

The **simultaneous-recording-servers** configuration attribute controls the number of simultaneous SIP dialogs that the Oracle Communications Session Border Controller establishes to the SRSs in the session recording group per communication session. For instance, if a session recording group contains 3 SRSs, and **simultaneous-recording-servers** is set to **2**, the recording agent initiates a SIP INVITE to the next two SRSs based on the session recording group strategy. In this way, duplicative recording sessions are instantiated, allowing for recording redundancy in multiple SRSs or within a session recording group.

Note:

The Oracle Communications Session Border Controller streams media to all SRSs. Each SRS chooses whether or not to ignore the media by returning a `recvonly`(receive only) media line. This permits an SRS to select specific media to record in the recording session, as well as determine whether or not to record the media.

The number of simultaneous recording servers does not dictate the number of recording devices required to be active for a communication session. If two SRSs exist in a session recording group and **simultaneous-recording-servers** is set to **2**, if at least one recording device to any of the servers completes, the recording server is treated as being established.

Load Balancing

The Oracle Communications Session Border Controller supports recording server load balancing across members of a session recording group using the following strategies:

Note:

SRS groups support “round-robin” and “hunt” strategies only.

[Round-robin]: The Oracle Communications Session Border Controller remembers the last SRS that was used. Each new recording session selects the next SRS in the session recording group. When simultaneous-recording-servers is greater than 1, the next n recording servers are selected from the session recording group.

[hunt]: The Oracle Communications Session Border Controller successively attempts to contact SRSs in the session recording group until a successful recording dialog is established with the SRS, starting from the first SRS in the session recording group. The Oracle Communications Session Border Controller attempts to contact each SRS in the session reporting group once. When contact is exhausted, the recording device is considered failed. A SIP failure (response greater than 399, timeout or TCP setup failure) causes the Oracle Communications Session Border Controller to attempt the next possible SRS. When simultaneous-recording-servers is greater than 1, the Oracle Communications Session Border Controller attempts to establish n recording devices in a hunting fashion.

Session Recording Group within Logical Remote Entities

Each logical remote entity (session-agent, realm-config and sip-interface) has a **session-recording-server attribute**. This attribute is a reference to a specific SRS configuration and can be used to specify a session recording group instead. If a session recording group is specified instead of an SRS, the session recording group name must be prefixed with "**SRG:**" followed by the session recording group name. This distinguishes between an SRS being referenced and a session recording group being referenced.

With SIPREC, if an SRS or session recording group is configured on both the ingress and egress logical remote entities, both the ingress and egress SRS/session recording groups are used. This means that the Oracle Communications Session Border Controller records the media between participants twice (or more) - once for the ingress recorders and once for the egress recorders.

If both the ingress and egress SRS/session recording group are the same, the Oracle Communications Session Border Controller makes an optimization and only records the media once. Even if the ingress session recording group is the same exact set of SRSs as the egress session recording group (but with a different name), the Oracle Communications Session Border Controller replicates media to both destinations. However, if the same set of SRSs has the exact same identifier, the Oracle Communications Session Border Controller sends media to one and not both SRSs.

Selective Recording

SIPREC defines a number of use cases for which the Oracle Communications Session Border Controller can record communication sessions. These use cases include the use of selective based recording. A **selective recording** is one in which a unique recording server is created per communication session.

Note:

The Oracle Communications Session Border Controller does not support persistent recording.

For SRSs using selective recording, recording servers are unique per session recording group. For each selective SRS in a session recording group, during the setup of a new communication session, the recording metadata is the same for each recording device. The SRC initiates a new

SIP INVITE to the SRS carrying the metadata for that new recording server. The recording agent terminates the SIP dialog at the time that the recording session ends.

The lifetime of a recording session extends beyond the lifetime of the recorded communication. The SRC (Oracle Communications Session Border Controller) re-uses the recording session ID in the metadata instead of creating a new ID for each recording.

High Availability (HA) Support

An Oracle Communications Session Border Controller using SIPREC supports HA in the network. The Oracle Communications Session Border Controller replicates all metadata states between the active and standby Oracle Communications Session Border Controllers. Any recording dialogs in progress do not survive the failover, but all calls in progress are preserved. Additionally, the recording dialogs are replicated as well to the failed over Oracle Communications Session Border Controller so that in-dialog SIP requests continue to function.

Each recorded communication session replicated to a single SRS counts as two calls instead of one. The Oracle Communications Session Border Controller creates two flows between the two participants and two additional flows to the SRS for each of the parent flows.

SIPREC Configuration Procedure

The following configuration example assumes the Oracle Communications Session Border Controller has the session recording license enabled on the Oracle Communications Session Border Controller. Changes to the call session recording configuration for SIPREC are dynamic. Active calls in progress remain unaffected by the configuration changes. New calls, however, utilize the changes after a **Save** and **Activate** of the configuration.

The following attributes must be configured:

- session-recording-server
- **session-recording-group** (for RSS or 3rd party SRS high availability (HA) only)

and at least one of the following attributes:

- realm-config
- session-agent
- sip-interface

Session-recording-server Attribute

To configure the session-recording-server attribute:

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-related objects.

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

3. Type **session-recording-server** and press Enter to access the session recording server-related attributes.

```
ACMEPACKET(session-router)# session-recording-server  
ACMEPACKET(session-recording-server)#
```

4. **name** — Enter a unique name for the session recording server. This name can be referenced when configuring realm-config, session-agent, and sip-interface. Valid values are alpha-numeric characters. Default is no value specified.

```
ACMEPACKET(session-recording-server)# name SRS1
```

5. **(optional) description** — Enter a description for the session recording server. Valid values are alpha-numeric characters. Default is no value specified.

```
ACMEPACKET(session-recording-server)# description <recording server name>
```

6. **realm** — Enter the realm for which the session recording server belongs. Valid values are alpha-numeric characters. Default is no value specified.

```
ACMEPACKET(session-recording-server)# realm <realm name>
```

 **Note:**

Oracle recommends that the session recording server be configured in its own realm.

7. **mode** — Enter the recording mode for the session recording server. Valid values are:

- **selective** (default) - Unique recording server created per communication session
- **persistent** - Not supported.

```
ACMEPACKET(session-recording-server)# recording-mode selective
```

8. **destination** — Enter the destination IP address with IP port (port specification is optional) that defines the SIP address (request URI) of the session recording server. Enter values in the format 0.0.0.0:<port number>. Default is no value specified.

```
ACMEPACKET(session-recording-server)# destination 172.34.2.3:5060
```

9. **port** — Enter the port number to contact the session recording server. Valid values are 1024 to 65535. Default is 5060.

10. **transport-method** — Enter the protocol that the session recording server uses to accept incoming packets from the session reporting client on the network. Default is DynamicTCP. Valid values are:

- "" - No transport method used. Same as leaving this parameter value blank.
- UDP - User Datagram Protocol (UDP) is used for transport method.
- UDP+TCP - UDP and Transmission Control Protocol (TCP) are used for transport method.
- DynamicTCP - One TCP connection for EACH session is used for the transport method.
- StaticTCP - Only one TCP connection for ALL sessions is used for the transport method. This option saves resource allocation (such as ports) during session initiation.
- DynamicTLS - One Transport Layer Security (TLS) connection for EACH session is used for the transport method.
- StaticTLS - Only one TLS connection for ALL sessions is used for the transport method. This option saves resource allocation (such as ports) during session initiation.
- DTLS - Datagram TLS is used for the transport method.
- TLS+DTLS - TLS and DTLS are used for the transport method.

- **StaticSCTP** - Only one Stream Control Transmission Protocol (SCTP) connection for ALL sessions is used for the transport method. This option saves resource allocation (such as ports) during session initiation.

```
ACMEPACKET(session-recording-server)# protocol UDP
```

11. Enter **done** to save the session recording configuration.

```
ACMEPACKET(session-recording-server)# done
```

12. Enter **exit** to exit the session-recording-server configuration.

```
ACMEPACKET(session-recording-server)# exit
```

13. Enter **exit** to exit the session-router configuration.

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

14. Enter **exit** to exit the configure mode.

```
ACMEPACKET(configure)# exit
```

15. Enter **save-config** to save the session recording configuration.

```
ACMEPACKET# save-config
```

16. Enter **activate-config** to activate the session recording configuration.

```
ACMEPACKET# activate-config
```

Session-recording-group Attribute (for HA only)

For environments that required high availability (HA) requirements, configure the **session-recording-group** attribute.

To configure the session-recording-group attribute and enable HA:

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-related objects.

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

3. Type **session-recording-group** and press Enter to access the session recording group-related attributes.

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

4. **name** — Enter a unique name for the session recording group that is a collection of one or more session recording servers. This name can be referenced when configuring realm-config, session-agent, and sip-interface. Valid values are alpha-numeric characters. Default is no value specified.

```
ACMEPACKET(session-recording-group)# name <SRG Group Name>
```

Note:

The name of the session recording group must be prefixed with SRG.

5. **(optional) description** — Enter a description for the session recording group. Valid values are alpha-numeric characters. Default is no value specified.

```
ACMEPACKET(session-recording-group)# description <Recording Group Name>
```

6. **session-recording-servers** — Enter the names of the session recording servers that belong to this session recording group. Valid values are alpha-numeric characters. Default is no value specified.

```
ACMEPACKET(session-recording-group)# session-recording-servers SRS1,SRS2
```

 **Note:**

You must enter multiple servers as values for the session-recording-servers attribute.

7. **strategy** — Enter the load balancing strategy that the session reporting client (Oracle Communications Session Border Controller) uses when sending recordings to the session reporting server. Valid values are:
- **Round-robin** (default) - The Oracle Communications Session Border Controller remembers the last SRS that was used. Each new recording session selects the next SRS in the session recording group. When simultaneous-recording-servers is greater than 1, the next n recording servers are selected from the session recording group.
 - **hunt** - The Oracle Communications Session Border Controller successively attempts to contact SRSs in the session recording group until a successful recording dialog is established with the SRS, starting from the first SRS in the session recording group. The Oracle Communications Session Border Controller attempts to contact each SRS in the session reporting group once. When contact is exhausted, the recording device is considered failed. A SIP failure (response greater than 399, timeout or TCP setup failure) causes the Oracle Communications Session Border Controller to attempt the next possible SRS. When simultaneous-recording-servers is greater than 1, the Oracle Communications Session Border Controller attempts to establish n recording devices in a hunting fashion.
 - **least busy** - For some 3rd party recording devices, the number of concurrent recording servers proves to be the most taxing for system resources. The Oracle Communications Session Border Controller tracks the number of recording servers active to a given SRS at any given time. It uses this information to determine which SRS would be the best candidate for the next RS. The SRS with the fewest number of active recording servers receives the next RS. If two or more SRSs in a session recording group currently have the same number of active recording servers, the SRS configured first in the session recording group takes precedence.
 - **lowest sustained rate** (fewest-setups-per-minute) - For some 3rd party recording servers, processing large amounts of sessions in a short amount of time proves to be the most taxing on their system's resources. The Oracle Communications Session Border Controller tracks the number of recording server setups over a sliding window of five minutes. The SRS within the session recording group with the fewest setups per the window of time is selected as the next candidate for receiving the recorded session. If two or more SRSs in a session recording group currently have the same value for setups in the given window of time, then the SRS configured first in the session recording group takes precedence.

```
ACMEPACKET(session-recording-group)# strategy round-robin
```

8. **simultaneous-recording-servers** — Enter the number of simultaneous SIP dialogs that the session reporting client (Oracle Communications Session Border Controller) establishes to the session reporting servers in the session reporting group per communication session. Valid values are **1** to **3**. Default is **0**.

```
ACMEPACKET(session-recording-group)# simultaneous-recording-servers 2
```

9. Enter **done** to save the session recording group configuration.

```
ACMEPACKET(session-recording-group)# done
```

10. Enter **exit** to exit the session recording group configuration.

```
ACMEPACKET(session-recording-group)# exit
```

11. Enter **exit** to exit the session-router configuration.

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

12. Enter **exit** to exit the configure mode.

```
ACMEPACKET(configure)# exit
```

13. Enter **save-config** to save the session recording group configuration.

```
ACMEPACKET# save-config
```

14. Enter **activate-config** to activate the session recording group configuration.

```
ACMEPACKET# activate-config
```

Realm-config Attribute

Use the following procedure to configure the realm-config attribute and enable session recording:

1. **session-recording-server** — Enter the name of the session-recording server or the session-recording-group in the realm associated with the session reporting client (Oracle Communications Session Border Controller). Valid values are alpha-numeric characters. Default is no value specified.

```
ACMEPACKET(realm-config)# session-recording-server <srs-name>
```

or

```
ACMEPACKET(realm-config)# session-recording-server SRG:<group-name>
```

Note:

The value for this attribute is the name you specified in Step 4 of the [Session-recording-server Attribute](#) or Step 4 of the [Session-recording-group Attribute \(for HA only\)](#). If specifying a session-recording-group, you must precede the group name with "SRG:".

2. **session-recording-required** — Enter whether you want a call to be accepted by the Oracle Communications Session Border Controller when recording is not available. The default value is **disabled**.
 - **Enabled** — Restricts call sessions from being initiated when a recording server is not available.

- **Disabled** (default) — Allows call sessions to initiate even when the recording server is not available.

 **Note:**

Oracle recommends that the **session-recording-required** parameter remain disabled.

3. **session-max-life-limit** — Enter the maximum interval in seconds before the SBC must terminate long duration calls. The value supercedes the value of **session-max-life-limit** in the **sip-interface** and **sip-config** configuration elements and is itself superceded by the value of **session-max-life-limit** in the **session-agent** configuration element. The default value is 0 (off/ignored).

test

Session-agent Attribute

To configure the session-agent attribute and enable session recording:

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

```
ORACLE# configure terminal
```

2. Type **session-router** and press Enter to access the session router-related objects.

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

3. Type **session-agent** and press Enter to access the session agent-related attributes.

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

4. **session-recording-server** — Enter the name of the session-recording server or the session-recording-group to apply to the session recording client (Oracle Communications Session Border Controller). Valid values are alpha-numeric characters. Default is no value specified.

```
ORACLE(session-agent)# session-recording-server <srs-name>
```

or

```
ORACLE(session-agent)# session-recording-server SRG:<group-name>
```

 **Note:**

The value for this attribute is the name you specified in Step 4 of the [Session-recording-server Attribute](#) or Step 4 of the [Session-recording-group Attribute \(for HA only\)](#). If specifying a session-recording-group, you must precede the group name with **SRG:**.

5. **session-recording-required** — Enter whether or not you want a call to be accepted by the Oracle Communications Session Border Controller if recording is not available. Valid values are:
 - **Enabled** - Restricts call sessions from being initiated when a recording server is not available.

- **Disabled** (default)- Allows call sessions to initiate even if the recording server is not available.

```
ORACLE(session-agent)# session-recording-required disabled
```

Note:

Oracle recommends that the session-recording-required parameter remain disabled.

6. Enter **exit** to exit the session agent configuration.

```
ORACLE(session-agent)# exit
```

7. Enter **exit** to exit the session router configuration.

```
ORACLE(session-router)# exit
```

8. Enter **exit** to exit the configure mode.

```
ORACLE(configure)# exit
```

9. Enter **save-config** to save the session agent configuration.

```
ORACLE# save-config
```

10. Enter **activate-config** to activate the session agent configuration.

```
ORACLE# activate-config
```

Sip-interface Attribute

To configure the sip-interface attribute and enable session recording:

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

```
ORACLE# configure terminal
```

2. Type **session-router** and press Enter to access the session router-related objects.

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

3. Type **sip-interface** and press Enter to access the SIP interface-related attributes.

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

4. **session-recording-server** — Enter the name of the session-recording server or the session-recording-group to apply to the SIP interface on the session recording client (Oracle Communications Session Border Controller). Valid values are alpha-numeric characters. Default is no value specified.

```
ORACLE(sip-interface)# se ss on-recording-server SRG:<session recording
server name or session-recording group name>
```

Note:

The value for this attribute is the name you specified in Step 4 of the [Session-recording-server Attribute](#) or Step 4 of the [Session-recording-group Attribute](#) (for HA only).

5. **session-recording-required** — Enter whether or not you want a call to be accepted by the Oracle Communications Session Border Controller if recording is not available. Valid values are:

- **Enabled** - Restricts call sessions from being initiated when a recording server is not available.
- **Disabled** (default)- Allows call sessions to initiate even if the recording server is not available.

```
ORACLE(sip-interface)# session-recording-required disabled
```

 **Note:**

Oracle recommends that the session-recording-required parameter remain disabled.

6. Enter **exit** to exit the SIP interface configuration.

```
ORACLE(sip-interface)# exit
```

7. Enter **exit** to exit the session router configuration.

```
ORACLE(session-router)# exit
```

8. Enter **exit** to exit the configure mode.

```
ORACLE(configure)# exit
```

9. Enter **save-config** to save the SIP interface configuration.

```
ORACLE# save-config
```

10. Enter **activate-config** to activate the SIP interface configuration.

```
ORACLE# activate-config
```

P-Asserted Identity and Diversion Headers in SIPREC Metadata

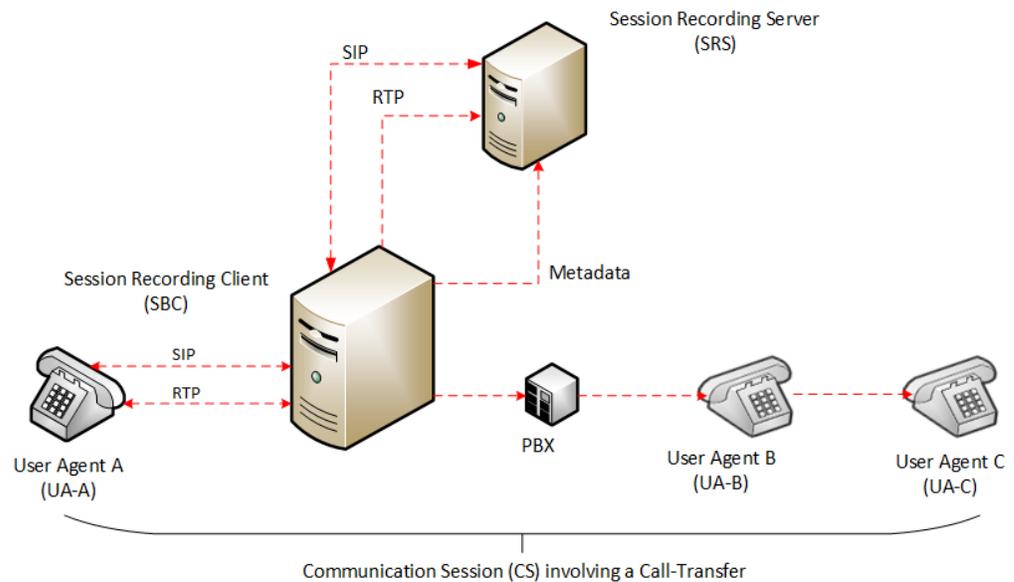
The Oracle Communications Session Border Controller supports some call transfer scenarios in which the contents of the P-Asserted-Identity, Diversion and History-info headers must be included in the SIPREC metadata for in-dialog requests (re-INVITE and UPDATE) as well as initial requests.

During a Communication Session (CS) between user-agents, Oracle Communications Session Border Controller updates the SRS, for in-dialog requests like Re-INVITE and UPDATE, in the Recording Session (RS). In call transfer scenarios, the participant information available in the headers is updated in the extension data under SIPREC Metadata. When the option **disable-re-invite-on-update** is configured in the session-agent, sip-interface, or real-config configuration elements, the SBC restricts the Re-INVITE to RS for the in-dialog UPDATE in CS.

You will have to configure the system with the SipHeaderExtensionMetadata.spl (SBC Processing Language) plugin for the **disable-re-invite-on-update** option to work. For more information, see Inserting SIP headers into SIPREC Metadata.

Consider a call transfer scenario between User Agent A (UA-A) and User Agent C (UA-C). The session will be recorded by Oracle Communications Session Border Controller acting as an SRC and an SRS. The SRS does not record the session with User Agent B. In order to record the session between UA-A and UA-C it is vital to correctly identify the User Agents in the

session. The system uses the headers in the metadata to reflect the identities of the participating agents.



The header information such as P-Asserted Identity, Diversion and History-info for INVITE/Re-INVITE and UPDATE cases will be updated in the extension data of participant xml element.

You can configure the option **disable-re-invite-on-update** to stop the Re-INVITE requests towards the Recording Session (RS) for any in-dialog UPDATE in Communication Session (CS). This option can be configured under a specified **sip-interface**, **realm-config**, or a **session-agent**.

```
ORACLE#(session-agent) options +disable-re-invite-on-update
```

Session-agent takes precedence over **realm-config**, and **realm-config** takes precedence over **sip-interface**.

Metadata Contents

The recording metadata contains a set of related elements which define the recording session. A recording session may contain zero or more communication sessions and/or communication session groups. A communication session represents a call instance; a communication session group represents a related group of communication sessions. A recording session is composed of a sequence of complex element types. Not all element types are required to describe a recording session initiated from the Oracle Communications Session Border Controller. The recording session XML schema defines the following element types:

- **dataMode** - partial or complete metadata description (required)
- **group** - a collection of related communication sessions
- **session** - a single communication session of two or more participants (required)
- **participant** - a SIP endpoint representation (required)
- **stream** - a media stream

- **extensiondata** - application specific data outside of the SIPREC scope.

The recording agent generates dataMode, session, participant, and stream elements. Extension data is attached to other elements within the metadata through the use of the parent attribute. The recording metadata is defined as a sequence of element types; therefore all associations between elements are represented as references to element identifiers.

The state of the metadata within a recording session reflects the state of the communication session(s) which is being recorded. SIPREC implements stop-times and reason codes when communication sessions end within a recording session. Once a communication session, participant, or media stream has been marked as 'stopped' and accepted by the SRS, the metadata item is removed from the current metadata state. In addition, media lines within the SDP or the recording session may be re-used/re-labeled for reuse if new communication sessions and media streams are created within the recording session.

The XML schema for the recording metadata is defined in the IETF draft RFC *draft-ram-siprec-metadata-format-02* [7].

The ACLI command to show recorded metadata is **show rec**. For more information on this command see the section, [Show rec](#).

Show Commands for Recording Sessions

The Oracle Communications Session Border Controller allows you to utilize the following **show** commands via the ACLI to display statistical information about recording sessions:

- show rec
- show rec redundancy

Show rec

The **show rec** command displays the count of all metadata objects in sessions managed by the recording agent. These statistics include metadata monitored over an active **period** of time and over a lifetime period (where lifetime totals reflect from the last reboot of the Oracle Communications Session Border Controller to the present time). The following example shows the use of this command.

1. Log into the Oracle Communications Session Border Controller as a User or Superuser.

```
ACMEPACKET> enable
ACMEPACKET(enable)#
```

2. Type **show rec** and press Enter to display the recording metadata statistics. The following output is an example of the show rec command.

```
ACMEPACKET(enable)# show rec
```

Show rec output

```
13:49:44-81645
Recording Agent Status      -- Period -- ----- Lifetime -----
                        Active  High  Total    Total  PerMax  High
Rec Sessions            0     1     1       1     1     1
Comm Groups             0     0     0       0     0     0
Comm Sessions           0     1     1       1     1     1
Media Streams           0     2     2       2     2     2
Participants            0     2     2       2     2     2
```

The following table describes the metadata objects in the show rec command output.

Object	Description
Rec Sessions	Number of recording sessions during an active period of time and over a lifetime period.
Comm Groups	Number of active communication session recording groups during an active period of time and over a lifetime period.
Comm Sessions	Number of active communication sessions during an active period of time and over a lifetime period.
Media Streams	Number of active media streams during an active period of time and over a lifetime period.
Participants	Total number of participants in session recordings during an active period of time and over a lifetime period.

Show rec redundancy

The **show rec redundancy** command displays information for session recording server statistics when the Oracle Communications Session Border Controller is configured for HA. These statistics include metadata monitored over an active period of time and over a lifetime period (where lifetime totals reflect from the last reboot of the Oracle Communications Session Border Controller to the present time) on both the primary and redundant Oracle Communications Session Border Controller. The following example shows the use of this command.

1. Log into the Oracle Communications Session Border Controller as a User or Superuser.

```
ACMEPACKET> enable
ACMEPACKET(enable)#
```

2. Type **show rec redundancy** and press Enter to display the session recording server statistics for Oracle Communications Session Border Controllers in HA mode. The following output is an example of the show rec redundancy command.

```
ACMEPACKET(enable)# show rec redundancy
```

Show rec redundancy output

Primary System

```
13:49:44-81645
Recording Agent Status      -- Period -- ----- Lifetime -----
Active High Total      Total PerMax High
Rec Sessions      0      1      1      1      1      1
Comm Groups      0      0      0      0      0      0
Comm Sessions    0      1      1      1      1      1
Media Streams    0      2      2      2      2      2
Participants     0      2      2      2      2      2
```

Redundant System

```
13:49:44-81646
Recording Agent Status      -- Period -- ----- Lifetime -----
Active High Total      Total PerMax High
Rec Sessions      0      1      1      1      1      1
Comm Groups      0      0      0      0      0      0
Comm Sessions    0      1      1      1      1      1
Media Streams    0      2      2      2      2      2
Participants     0      2      2      2      2      2
```

The following table describes the session recording server statistics in the **show rec redundancy** command output.

Object	Description
Rec Sessions	Number of recording sessions during an active period of time and over a lifetime period.
Comm Groups	Number of active communication session recording groups during an active period of time and over a lifetime period.
Comm Sessions	Number of active communication sessions during an active period of time and over a lifetime period.
Media Streams	Number of active media streams during an active period of time and over a lifetime period.
Participants	Total number of participants in session recordings during an active period of time and over a lifetime period.

Inserting SIP Headers into SIPREC Metadata

The SIPREC Extension Data Enhancements SPL provides additional header information in the originating SIP messages metadata sent to the Interactive Session Recorder. With this SPL, you can introduce more options for recording policy decisions when using the SIPREC feature of the Oracle Communications Session Border Controller. The enhanced metadata also allows for the realm-id to be used as an indicator of the recording account. The SPL also provides configurable values that collect additional header information to store in the metadata.

When the SPL is configured, the SIPREC Extension Data Enhancements SPL is only triggered upon INVITE/UPDATE requests, and stores the additional header information in the metadata that is sent to the Net-Net Interactive Session Recorder (NN-ISR). Metadata is a XML MIME attachment that describes recording details to the Net-Net ISR.

By default, the **Extension-Headers** SPL option collects only the Request-URI in a received INVITE. You can store additional header information by configuring the SPL with additional attributes in the **spl-options** under the global **spl-config**. The values must be in a comma separated list enclosed in double quotation marks. For example:

```
Extension-Headers="P-Asserted-Identity,Diversion"
```

This configuration of the **Extension-Headers** option adds the originating Request-URI along with all P-Asserted-Identity and Diversion-Headers into the participant section of the metadata.

You can configure the **LRE-Identifier** SPL option to add an identifier of the logical remote entity (LRE) that triggered the recording to the <apkt:realm> element of the extension metadata. When configured with a value added, the value appears in place of the identifier. When configured without a value, the identifier of the logical remote entity is used. For example, session-agent will be the hostname, realm-config will be the realm, and sip-interface will be the realm name.

Note:

Both options are required for the SPL to function properly.

Sample Metadata

The sample below shows metadata with new extension data added by the SIPREC Extension Data Enhancements SPL (New metadata appears in bold):

```

<?xml version='1.0' encoding='UTF-8'?>
  <recording xmlns='urn:ietf:params:xml:ns:recording'>
    <datamode>complete</datamode>
    <session id="BYiC7uSZQGN3VQdzWI1HWw==">
      <associate-time>2012-06-26T13:44:13</associate-time>
    </session>
    <participant id="hq18GJs3TtJdhjPsfPNV8A=="
      session="BYiC7uSZQGN3VQdzWI1HWw==">
      <nameID aor="sip:sipp@192.168.101.1">
        <name>sipp</name>
      </nameID>
      <send>aD50KX+LTvxNzASg+/GQTg==</send>
      <associate-time>2012-06-26T13:44:13</associate-time>
      <extensiondata xmlns:apkt="http://acmepacket.com/siprec/
extensiondata">
        <apkt:callingParty>true</apkt:callingParty>
        <apkt:request-uri>sip:service@192.168.101.13:5060 </
apkt:request-uri>
        <apkt:in-realm>net192</apkt:in-realm>
        <apkt:header label=P-Asserted-Identity>
          <value>sip:mike@acme.com</value>
          <value>sip:bob@cisco.com</value></apkt:header>
        <apkt:header label=Diversion>
          <value>&lt;sip:jojo@divert.com&gt;;happy=days;green=envy</value>
          <value>&lt;sip:bebe@MediaTen.net&gt;;green=monster;go=carts</value>
            <value>&lt;tel:
+8675309;night=mare&gt;;gear=head;green=monitor</value></apkt:header>
        </extensiondata>
      </participant>
      <participant id="Ki6WEUi4TPRUPLtEaEhA7Q=="
        session="BYiC7uSZQGN3VQdzWI1HWw==">
        <nameID aor="sip:service@192.168.101.13">
          <name>sut</name>
        </nameID>
        <send>f9NDVhyMTul+ePlM2SceQA==</send>
        <associate-time>2012-06-26T13:44:13</associate-time>
        <extensiondata xmlns:apkt="http://acmepacket.com/siprec/
extensiondata">
          <apkt:callingParty>>false</apkt:callingParty>
        </extensiondata>
      </participant>
      <stream id="aD50KX+LTvxNzASg+/GQTg==" session="BYiC7uSZQGN3VQdzWI1HWw==">
        <label>65804</label>
        <mode>separate</mode>
        <associate-time>2012-06-26T13:44:13</associate-time>
      </stream>
      <stream id="f9NDVhyMTul+ePlM2SceQA==" session="BYiC7uSZQGN3VQdzWI1HWw==">
        <label>65805</label>
        <mode>separate</mode>
        <associate-time>2012-06-26T13:44:13</associate-time>
      </stream>
    </recording>

```

Configure SIP Headers for SIPREC Metadata

To get more detailed information about a recorded session, you can add more SIP headers within the SIPREC metadata by way of the **Extension-Headers** option. The default behavior stores only the Request-URI and realm-id.

You must configure the **Extension-Headers** option at the global level under **spl-config** because the session-agent, realm-config, and sip-interface configurations do not recognize the option. The first time you configure one or more extension headers, you need only to save and activate the configuration for the system to recognize the extension headers. When you modify the existing SPL extension header list you need to save and activate the configuration, and reboot the system for the changes to take effect. Real Time Configuration (RTC) does not apply to extension header options.

1. Type **spl-options +Extension-Headers="**<value>**"** , where **<value>** is the additional header information to store, and press Enter.

```
ACMESYSTEM(spl-config)# spl-options +Extension-Headers="P-Asserted-Identity,Diversion"
```

2. Type **done** to save the configuration.

SIPREC Recording Session Refresh

The Oracle Communications Session Border Controller provides for periodic in-dialog OPTIONS request/response exchanges to test the availability of the recording session dialog.

As shown below, and previously explained, establishment of a SIPREC session involves the creation of three distinct dialogs: a recording session dialog between the SIPREC client and SIPREC server, a SIP dialog between the calling party (A) and the SBC (acting as a B2BUA), and another SIP dialog between the SBC (acting as a B2BUA) and the called party (B). The two SIP dialogs are viewed conceptually as a single communications session dialog.

Both the call recording and the communication session dialogs are triggered by an INVITE at the SIPREC client. Upon reception, the recording client (the SBC) buffers the original INVITE, and sends a copy if it, along with call-related meta-data to the recording server. A 200 OK response from the server establishes the recording session dialog that will carry replicated call content; signalling information is conveyed by the metadata in the recording dialog. After establishing the recording session, the client transmits the buffered INVITE to the called party to begin establishing the communications session dialog.

In the absence of a 200 OK response, the recording client is required to reject the offered INVITE with a 503 (Service Unavailable) SIP error code and an accompanying Reason header indicating that the recording session could not be established.

After establishing the recording session dialog, the client does not monitor its connection state. Its only opportunity to verify the state occurs after the termination of the communications session dialog, which triggers a client-originated BYE to the recording server. Reception of a 200 OK/ACK confirms the persistence of the recording session, while the lack of acknowledgement indicates the failure of the recording session at some unspecified point in time.

This failure to identify a prematurely terminated recording session has caused service providers and some regulatory agencies to require timely recognition of a failed call recording dialog.

Release S-CZ7.2.0, and later releases, address this requirement by providing for the exchange of periodic in-dialog OPTIONS request/response exchanges to test the availability of the recording session dialog.

Timer_B

Section 17.1.1.2 of RFC 3261, SIP: Session Initiation Protocol describes a `TIMER_B`, the SIP INVITE transaction timer, which specifies the maximum interval between an INVITE request and response. The RFC suggests a default value of 32 seconds. Depending on the installed release version, two CLI commands, `trans-expire` and `initial-inv-trans-expire`, provide user control over the `TIMER_B` value at both the SIP global level and the SIP interface level. Use these commands to adjust `TIMER_B` values if required.

OPTIONS Request/Response

Availability of the recording session is tested by an in-dialog SIP OPTIONS request/response exchange initiated by the call recording client and completed by the call recording server. If the OPTIONS exchange is enabled, two timers are set when the recording session dialog is established: a refresh-timer that specifies the interval between OPTIONS requests sent by the call recording client, and a response-timer that specifies the maximum interval between the OPTIONS request and the OPTIONS response. The response-timer is set to the smaller of the configured `TIMER_B` or refresh-timer values.

Expiration of the refresh-timer results in the transmission of an OPTIONS request to the call recording server. The server, in turn, must reply with a 200 OK OPTIONS response prior to the expiration of the response-timer. In the event of a positive response, the call recording client restarts the refresh-timer, and re-issues the OPTIONS request when the timer next expires.

In the event that the call recording server response is non-positive, that is either

1. not received prior to the expiration of the response timer, or
2. a non-2xx OPTIONS response

the SBC terminates the session recording dialog and records the termination in the event log.

Typically, a SIPREC environment contains multiple recorders configured to record a communications session dialog, consequently the communications session should be terminated only when all recording dialogs in the call recording server have terminated. At that point the communications session dialog is torn down with a 503 (Service Unavailable) error report.

Recording Session Refresh Configuration

Use the following procedure to enable a SIP OPTIONS request/response mechanism used to detect a prematurely terminated call recording session dialog.

1. **session-recording-required**—ensure that this parameter is enabled.
This is a prerequisite for the provision of recording session refresh services.
2. **refresh-interval**—use this parameter to both enable the SIP OPTIONS request/response mechanism, and to assign a value to the refresh-timer, which measures the maximum allowed interval (in seconds) between the OPTIONS request sent by the call-recording client and the OPTIONS response returned by the call-recording server.
By default, refresh-interval is set to 0, which disables detection of a failed recording session dialog.
Assignment of any non-zero value enables detection and sets the allowable interval between OPTIONS requests and responses.

Consult local policy and regulatory requirements when specifying the refresh-interval value.

```
ACMEPACKET(session-router)# refresh-interval 12
ACMEPACKET(session-recording-server)#
```

Codec Negotiation

In a SIPREC environment, it is assumed that the recording ecosystem provides transcoding media servers for which media calls can be redirected to, relieving the issue of codec matching from the recording servers. However, if transcoding media servers are not provided, the responsibility for transcoding falls on the recording server or the recording client in a SIPREC environment. The Oracle Communications Session Border Controller/SRC is required to impose some policy decisions on the codec negotiation between the three, or more, end-points. Specifically, the codec negotiation between the two participants and the recording server is subject to additional policy actions.

The SDP answer from the SRS may not agree with the media flows established in the communication session between UA-A and UA-B. If UA-A and UA-B agree to use G729, yet the SRS's answer indicates no support for G729, the SRS is then unable to interpret the media streams. The SDP offer forwarded to the called party (in this case UA-B) limits the codec choices to those supported by the SRS.

Note:

The recording agent forwards the original codec offer to the SRS prior to sending the invite to the UA-B. The SRS responds with the SDP answer, indicating the codec list most desirable to the SRS. The codec list in the answer is then forwarded to UA-B. This allows three parties in a conference call to participate in the negotiation of the codecs among the supported formats only.

SIPREC Call Flows

This section provides examples of call flow scenarios that can occur in a SIPREC environment. SIP recording call flow examples include:

For Selective Recording:

- Normal Call (recording required)
- Normal Call (recording not required)
- Early Media Call (recording not required)
- REFER Pass-Through Call (REFER handled by User Agent)
- REFER Call (REFER handled by the Oracle Communications Session Border Controller)
- SRS Indicates Busy in Call (recording not required)



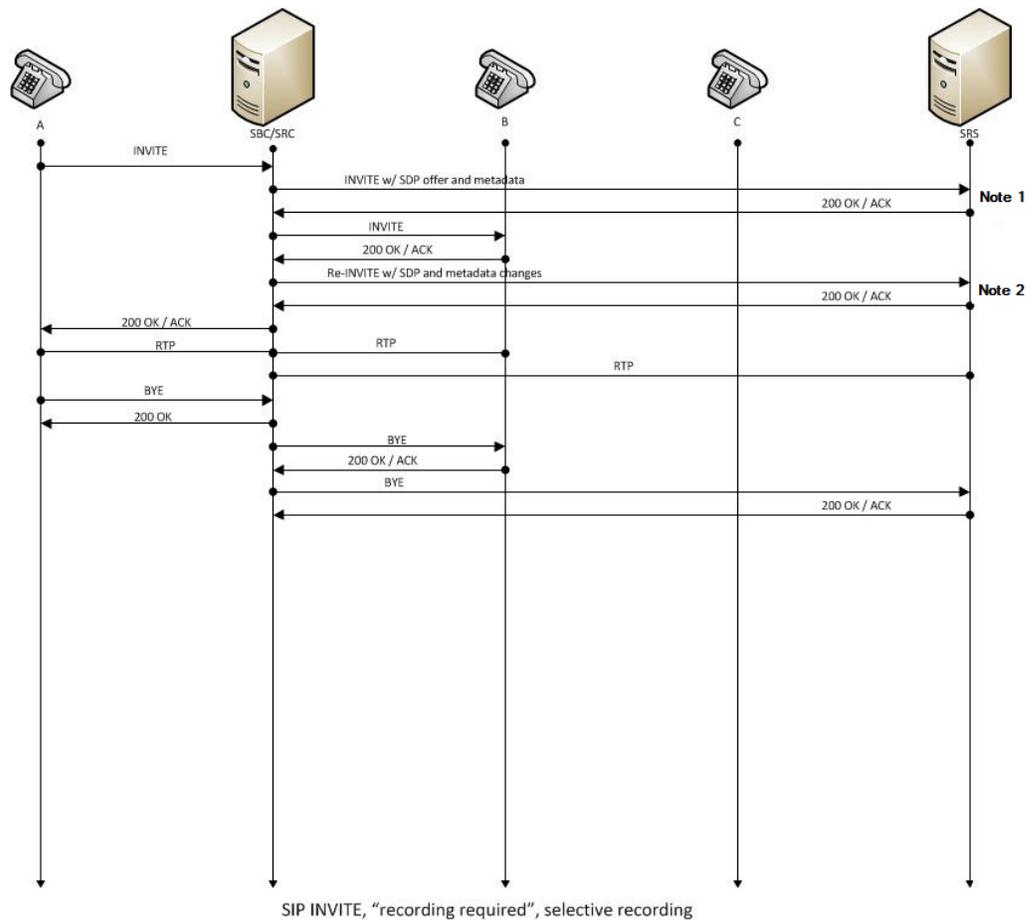
Note:

REFER is a SIP method indicating that the recipient (identified by the Request-URI) should contact a third party using the contact information provided in the request.

Selective Recording

Normal Call (recording required)

The following illustration shows a normal call using selective recording with recording required. For SDP and Metadata information in Notes 1 and 2, see [Sample SDP and Metadata](#).



I

Call Flow Description

① UA-A sends INVITE to Oracle Communications Session Border Controller.

⑩ RTP stream initiated between Oracle Communications Session Border Controller and UA-B.

Call Flow Description

② Oracle Communications Session Border Controller forwards INVITE with SDP and metadata to SRS.	⑪ RTP stream initiated between Oracle Communications Session Border Controller and SRS.
③ SRS responds with OK to Oracle Communications Session Border Controller.	⑫ UA-A sends BYE to Oracle Communications Session Border Controller.
④ Oracle Communications Session Border Controller sends INVITE to UA-B.	⑬ Oracle Communications Session Border Controller responds with OK to UA-A.
⑤ UA-B responds with OK to Oracle Communications Session Border Controller.	⑭ Oracle Communications Session Border Controller sends BYE to Oracle Communications Session Border Controller.
⑥ Oracle Communications Session Border Controller sends re-INVITE with SDP and metadata changes to SRS.	⑮ Oracle Communications Session Border Controller responds with OK to UA-A.
⑦ SRS responds with OK to Oracle Communications Session Border Controller.	⑯ Oracle Communications Session Border Controller sends BYE to UA-B.
⑧ Oracle Communications Session Border Controller forwards OK response to UA-A.	⑰ UA-B responds with OK to Oracle Communications Session Border Controller.
⑨ RTP stream initiated between UA-A and Oracle Communications Session Border Controller.	⑱ Oracle Communications Session Border Controller sends BYE to SRS.
	⑲ SRS responds with OK to Oracle Communications Session Border Controller.

Sample SDP and Metadata

The following sample SDP and Metadata pertain to Notes 1 and 2 in the previous Call Flow diagram.

```
--[Note 1]-----
Content-Type: application/sdp
v=0
o=- 171 213 IN IP4 10.0.0.2
s=-
c=IN IP4 10.0.0.1
t=0 0
m=audio 6000 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=label:1

Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session
<?xml version='1.0' encoding='UTF-8'?>
<recording xmlns='urn:ietf:params:xml:ns:recording'>
  <dataMode>complete</dataMode>
  <session id="urn:uuid:79b2fcd8-5c7f-455c-783f-db334e5d57d0">
    <start-time>2011-06-27T17:03:57</start-time>
  </session>
  <participant id="urn:uuid:10ac9063-76b7-40bb-4587-08ba290d7327"
session="urn:uuid:79b2fcd8-5c7f-455c-783f-db334e5d57d0">
    <aor>sip:sipp@168.192.24.40</aor>
    <name>sipp </name>
    <send>urn:uuid:07868c77-ef8e-4d6f-6dd5-a02ff53a1329</send>
    <start-time>2011-06-27T17:03:57</start-time>
  </participant>
<participant id="urn:uuid:797c45f5-e765-4b12-52b0-d9be31138529"
```

```

session="urn:uuid:79b2fcd8-5c7f-455c-783f-db334e5d57d0">
  <aor>sip:service@168.192.24.60</aor>
  <name>sut </name>
</participant>
<stream id="urn:uuid:4a72aled-abb2-4d7c-5f4d-6d4c36e2d4ec"
session="urn:uuid:79b2fcd8-5c7f-455c-783f-db334e5d57d0">
  <mode>separate</mode>
  <start-time>2011-06-27T17:03:57</start-time>
  <label>1</label>
</stream>
</recording>

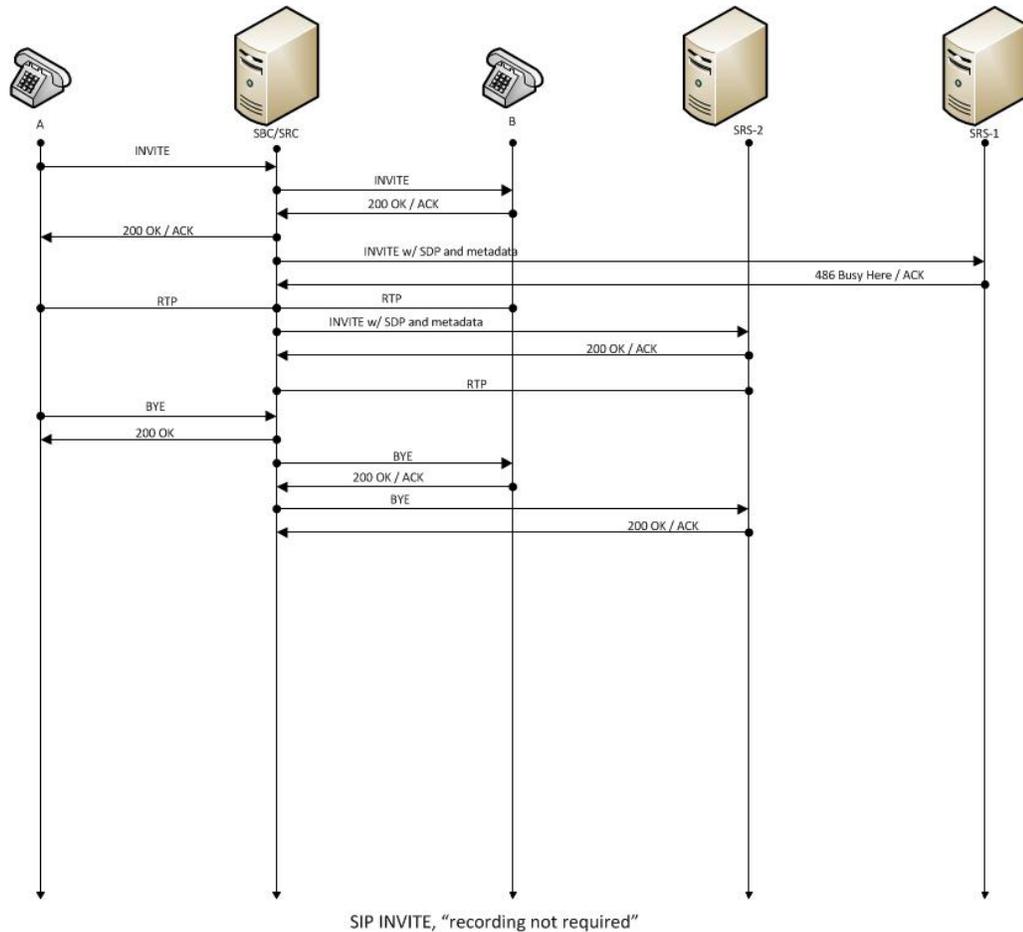
--[Note 2]-----
Content-Type: application/sdp
v=0
o=- 171 213 IN IP4 10.0.0.2
s=-
c=IN IP4 10.0.0.1
t=0 0
m=audio 6000 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=label:1
m=audio 6002 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=label:2

Content-Type: application/rs-metadata+xml
Content-Disposition: recording-session
<?xml version='1.0' encoding='UTF-8'?>
<recording xmlns='urn:ietf:params:xml:ns:recording'>
  <dataMode>partial</dataMode>
  <session id="urn:uuid:79b2fcd8-5c7f-455c-783f-db334e5d57d0">
    <start-time>2011-06-27T17:03:57</start-time>
  </session>
  <participant id="urn:uuid:797c45f5-e765-4b12-52b0-d9be31138529"
session="urn:uuid:79b2fcd8-5c7f-455c-783f-db334e5d57d0">
  <aor>sip:service@168.192.24.60</aor>
  <name>sut </name>
  <send>urn:uuid:4a72aled-abb2-4d7c-5f4d-6d4c36e2d4ec</send>
  <start-time>2011-06-27T17:03:58</start-time>
</participant>
  <stream id="urn:uuid:07868c77-ef8e-4d6f-6dd5-a02ff53a1329"
session="urn:uuid:79b2fcd8-5c7f-455c-783f-db334e5d57d0">
  <mode>separate</mode>
  <start-time>2011-06-27T17:03:58</start-time>
  <label>2</label>
</stream>
</recording>

```

Normal Call (recording not required)

The following illustration shows a normal call using selective recording with recording optional.

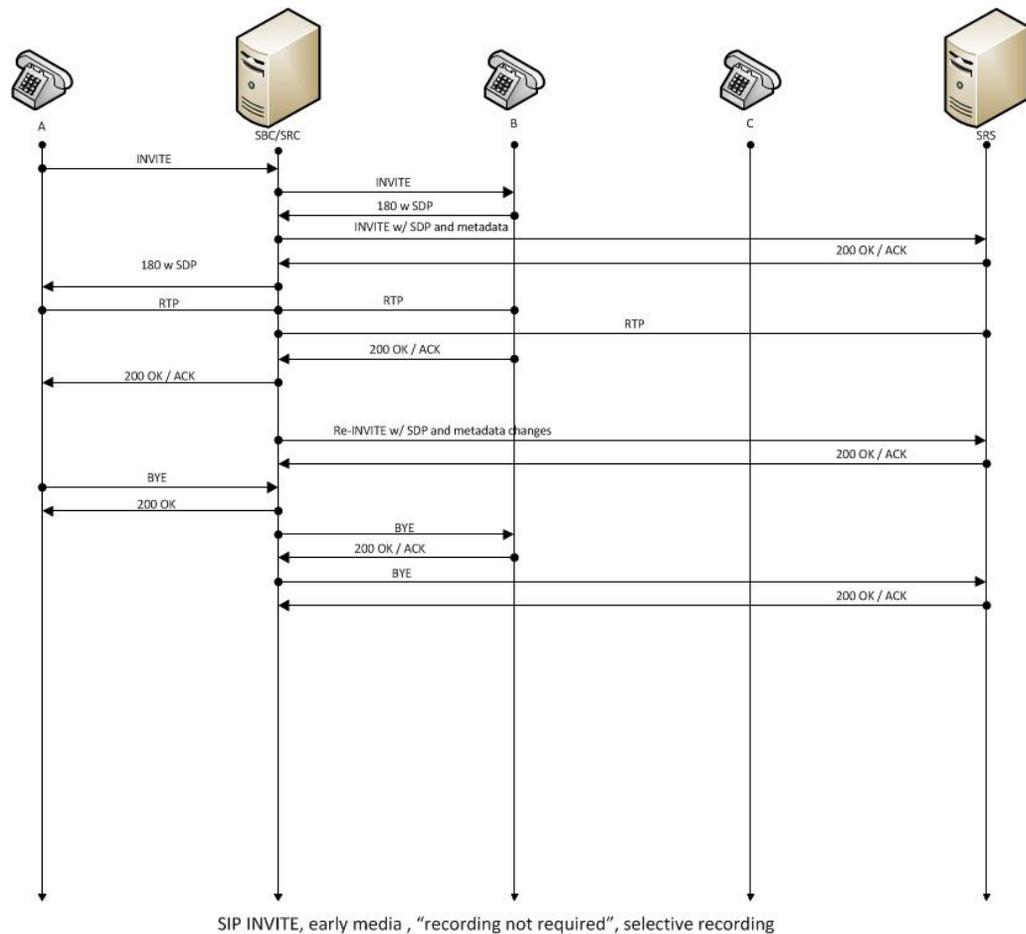


Call Flow Description

- | | |
|--|--|
| <ol style="list-style-type: none"> ① UA-A sends INVITE to Oracle Communications Session Border Controller. ② Oracle Communications Session Border Controller forwards INVITE to UA-B. ③ UA-B responds with OK to Oracle Communications Session Border Controller. ④ Oracle Communications Session Border Controller forwards OK response to UA-A. ⑤ Oracle Communications Session Border Controller sends INVITE with SDP and metadata to SRS. ⑥ SRS responds with OK to Oracle Communications Session Border Controller. ⑦ RTP stream initiated between UA-A, Oracle Communications Session Border Controller, and UA-B. | <ol style="list-style-type: none"> ⑧ RTP stream initiated between Oracle Communications Session Border Controller and SRS. ⑨ UA-A sends BYE to Oracle Communications Session Border Controller. ⑩ Oracle Communications Session Border Controller responds with OK to UA-A. ⑪ Oracle Communications Session Border Controller sends BYE to UA-B. ⑫ UA-B responds with OK to Oracle Communications Session Border Controller. ⑬ Oracle Communications Session Border Controller sends BYE to SRS. ⑭ SRS responds with OK to Oracle Communications Session Border Controller. |
|--|--|

Early Media Call (recording not required)

The following illustration shows an early media call using selective recording with recording optional.



Call Flow Description

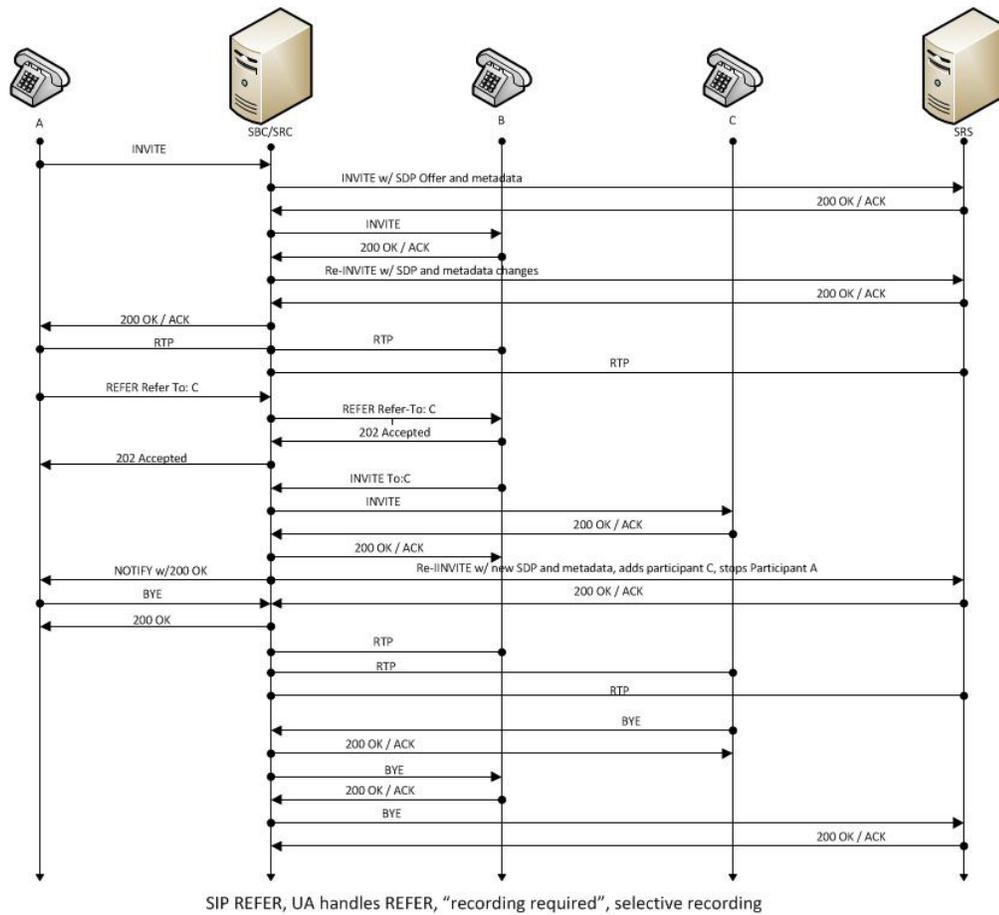
- ① UA-A sends INVITE to Oracle Communications Session Border Controller.
- ② Oracle Communications Session Border Controller forwards INVITE to UA-B.
- ③ UA-B sends 180 and SDP to Oracle Communications Session Border Controller.
- ④ Oracle Communications Session Border Controller sends INVITE with SDP and metadata to SRS.
- ⑤ SRS responds with OK to Oracle Communications Session Border Controller.
- ⑥ Oracle Communications Session Border Controller sends 180 with SDP to UA-A.
- ⑦ RTP stream initiated between Oracle Communications Session Border Controller and UA-A.
- ⑧ RTP stream initiated between Oracle Communications Session Border Controller and UA-B.
- ⑨ UA-B responds with OK to Oracle Communications Session Border Controller.
- ⑩ UA-B responds with OK to Oracle Communications Session Border Controller.
- ⑪ Oracle Communications Session Border Controller forwards OK to UA-A.
- ⑫ Oracle Communications Session Border Controller sends re-INVITE with SDP and metadata changes to SRS.
- ⑬ SRS responds with OK to Oracle Communications Session Border Controller.
- ⑭ UA-A sends BYE to Oracle Communications Session Border Controller.
- ⑮ Oracle Communications Session Border Controller responds with OK to UA-A.
- ⑯ Oracle Communications Session Border Controller sends BYE to UA-B.
- ⑰ UA-B responds with OK to Oracle Communications Session Border Controller.

Call Flow Description

- ⑨ RTP stream initiated between Oracle Communications Session Border Controller and SRS.
- ⑩ Oracle Communications Session Border Controller sends BYE to SRS.
- ⑪ SRS responds with OK to Oracle Communications Session Border Controller.
- ⑫ Oracle Communications Session Border Controller sends INVITE to UA-B.
- ⑬ UA-B responds with 200 OK to Oracle Communications Session Border Controller.
- ⑭ Oracle Communications Session Border Controller sends Re-INVITE with SDP and metadata changes to SRS.
- ⑮ SRS responds with 200 OK to Oracle Communications Session Border Controller.
- ⑯ Oracle Communications Session Border Controller sends REFER Refer-To: C to UA-A.
- ⑰ UA-A responds with 202 Accepted to Oracle Communications Session Border Controller.
- ⑱ Oracle Communications Session Border Controller sends INVITE To: C to UA-C.
- ⑲ UA-C responds with 200 OK to Oracle Communications Session Border Controller.
- ⑳ Oracle Communications Session Border Controller sends Re-INVITE with new SDP and metadata, adds participant C, stops Participant A to SRS.
- ㉑ SRS responds with 200 OK to Oracle Communications Session Border Controller.
- ㉒ Oracle Communications Session Border Controller sends NOTIFY w/200 OK to UA-A.
- ㉓ UA-A responds with BYE to Oracle Communications Session Border Controller.
- ㉔ Oracle Communications Session Border Controller sends 200 OK to UA-A.
- ㉕ Oracle Communications Session Border Controller sends RTP to UA-B.
- ㉖ Oracle Communications Session Border Controller sends RTP to UA-C.
- ㉗ Oracle Communications Session Border Controller sends BYE to UA-C.
- ㉘ UA-C responds with 200 OK to Oracle Communications Session Border Controller.
- ㉙ Oracle Communications Session Border Controller sends BYE to UA-B.
- ㉚ UA-B responds with 200 OK to Oracle Communications Session Border Controller.
- ㉛ Oracle Communications Session Border Controller sends BYE to SRS.
- ㉜ SRS responds with 200 OK to Oracle Communications Session Border Controller.

REFER Pass-Through Call (REFER handled by User Agent)

The following illustration shows a REFER pass-through call using selective recording and the User Agent (UA) handling the REFER on the call. Recording is required in this call flow.



Call Flow Description

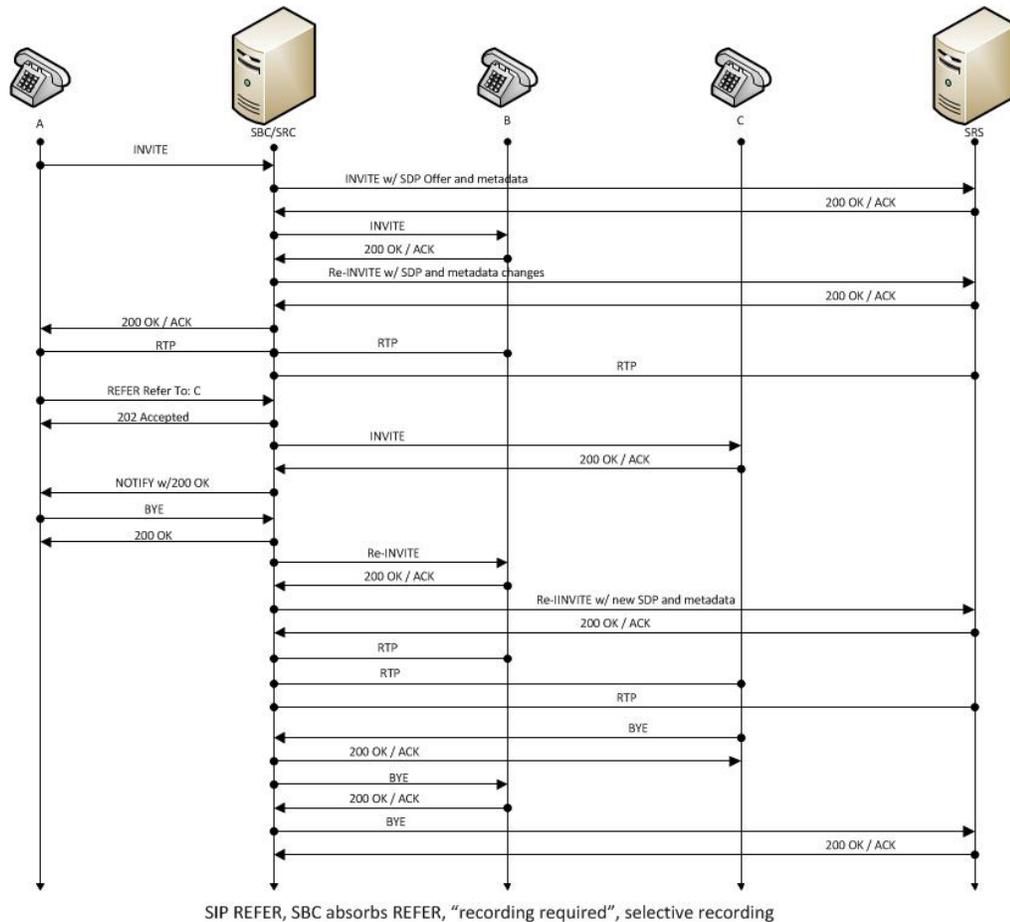
- 1 - UA-A sends INVITE to Oracle Communications Session Border Controller.
- 2 - Oracle Communications Session Border Controller forwards INVITE with SDP Offer and metadata to SRS.
- 3 - SRS responds with 200 OK to Oracle Communications Session Border Controller.
- 4 - Oracle Communications Session Border Controller forwards 200 OK to UA-A.
- 5 - Oracle Communications Session Border Controller sends RTP to UA-B.
- 6 - Oracle Communications Session Border Controller sends RTP to UA-C.
- 7 - Oracle Communications Session Border Controller sends BYE to UA-C.
- 8 - UA-C responds with 200 OK to Oracle Communications Session Border Controller.
- 9 - Oracle Communications Session Border Controller sends BYE to UA-B.
- 10 - UA-B responds with 200 OK to Oracle Communications Session Border Controller.
- 11 - Oracle Communications Session Border Controller sends BYE to SRS.
- 12 - SRS responds with 200 OK to Oracle Communications Session Border Controller.
- 13 - Oracle Communications Session Border Controller sends INVITE to UA-B.
- 14 - UA-B responds with 200 OK to Oracle Communications Session Border Controller.
- 15 - Oracle Communications Session Border Controller sends Re-INVITE with SDP and metadata changes to SRS.
- 16 - SRS responds with 200 OK to Oracle Communications Session Border Controller.
- 17 - Oracle Communications Session Border Controller sends REFER Refer-To: C to UA-A.
- 18 - UA-A responds with 202 Accepted to Oracle Communications Session Border Controller.
- 19 - Oracle Communications Session Border Controller forwards OK response to UA-B.
- 20 - Oracle Communications Session Border Controller sends INVITE To: C to UA-C.
- 21 - UA-C responds with 200 OK to Oracle Communications Session Border Controller.
- 22 - Oracle Communications Session Border Controller sends Re-INVITE with new SDP and metadata, adds participant C, stops Participant A to SRS.
- 23 - SRS responds with 200 OK to Oracle Communications Session Border Controller.
- 24 - Oracle Communications Session Border Controller sends NOTIFY w/200 OK to UA-A.
- 25 - UA-A responds with BYE to Oracle Communications Session Border Controller.
- 26 - Oracle Communications Session Border Controller sends 200 OK to UA-A.
- 27 - Oracle Communications Session Border Controller sends RTP to UA-B.
- 28 - Oracle Communications Session Border Controller sends RTP to UA-C.
- 29 - Oracle Communications Session Border Controller sends BYE to UA-C.
- 30 - UA-C responds with 200 OK to Oracle Communications Session Border Controller.
- 31 - Oracle Communications Session Border Controller sends BYE to UA-B.
- 32 - UA-B responds with 200 OK to Oracle Communications Session Border Controller.
- 33 - Oracle Communications Session Border Controller sends BYE to SRS.
- 34 - SRS responds with 200 OK to Oracle Communications Session Border Controller.

Call Flow Description

3 - SRS responds with OK to Oracle Communications Session Border Controller.	20 - Oracle Communications Session Border Controller sends NOTIFY with OK response to UA-A.
4 - Oracle Communications Session Border Controller sends INVITE to UA-B.	21 - Oracle Communications Session Border Controller sends re-INVITE to SRS with new SDP and metadata, adds participant C, stops participant A .
5 - UA-B responds with OK to Oracle Communications Session Border Controller.	22 - SRS responds with OK to Oracle Communications Session Border Controller.
6 - Oracle Communications Session Border Controller sends re-INVITE with SDP and metadata changes to SRS.	23 - UA-A sends BYE to Oracle Communications Session Border Controller.
7 - SRS responds with OK to Oracle Communications Session Border Controller.	24 - Oracle Communications Session Border Controller responds with OK to UA-A.
8 - Oracle Communications Session Border Controller forwards OK response to UA-A.	25 - Oracle Communications Session Border Controller responds with OK to UA-A.
9 - RTP stream initiated between UA-A and Oracle Communications Session Border Controller.	26 - RTP stream initiated between Oracle Communications Session Border Controller and UA-B.
10 - RTP stream initiated between Oracle Communications Session Border Controller and UA-B.	27 - RTP stream initiated between Oracle Communications Session Border Controller and UA-C.
11 - RTP stream initiated between Oracle Communications Session Border Controller and SRS.	28 - RTP stream initiated between Oracle Communications Session Border Controller and SRS.
12 - UA-A sends REFER-TO: C to Oracle Communications Session Border Controller.	29 - UA-C sends BYE to Oracle Communications Session Border Controller.
13 - Oracle Communications Session Border Controller forwards REFER-TO: C to UA-B.	30 - Oracle Communications Session Border Controller responds with OK to UA-C.
14 - UA-B responds with 202 ACCEPTED to Oracle Communications Session Border Controller.	31 - Oracle Communications Session Border Controller sends BYE to UA-B.
15 - Oracle Communications Session Border Controller forwards 202 ACCEPTED to UA-A.	32 - UA-B responds with OK to Oracle Communications Session Border Controller.
16 - UA-B sends INVITE TO: C to Oracle Communications Session Border Controller.	33 - Oracle Communications Session Border Controller sends BYE to SRS
17 - Oracle Communications Session Border Controller sends INVITE to UA-C.	34 - SRS responds with OK to Oracle Communications Session Border Controller.

REFER Call (REFER handled by Oracle Communications Session Border Controller)

The following illustration shows a call using selective recording and the Session Border Controller (Oracle Communications Session Border Controller) handling the REFER on the call. Recording is required in this call flow.



Call Flow Description

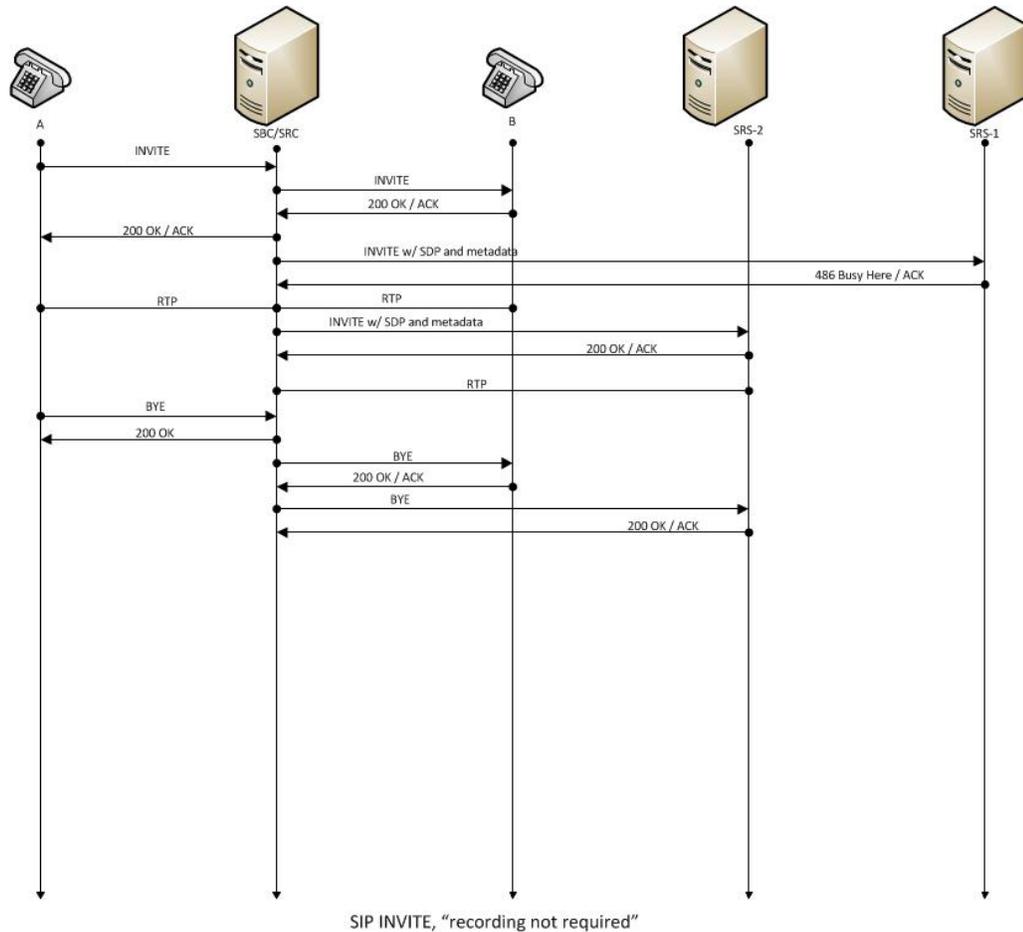
- | | |
|---|--|
| <p>1 - UA-A sends INVITE to Oracle Communications Session Border Controller.</p> <p>2 - Oracle Communications Session Border Controller forwards INVITE with SDP Offer and metadata to SRS.</p> <p>3 - SRS responds with OK to Oracle Communications Session Border Controller.</p> <p>4 - Oracle Communications Session Border Controller sends INVITE to UA-B.</p> <p>5 - UA-B responds with OK to Oracle Communications Session Border Controller.</p> <p>6 - Oracle Communications Session Border Controller sends re-INVITE with SDP and metadata changes to SRS.</p> <p>7 - SRS responds with OK to Oracle Communications Session Border Controller.</p> <p>8 - Oracle Communications Session Border Controller forwards OK response to UA-A.</p> | <p>16 - Oracle Communications Session Border Controller sends NOTIFY with OK response to UA-A.</p> <p>17 - UA-A sends BYE to Oracle Communications Session Border Controller.</p> <p>18 - Oracle Communications Session Border Controller responds with OK to UA-A.</p> <p>19 - Oracle Communications Session Border Controller sends re-INVITE to UA-B.</p> <p>20 - UA-B responds with OK to Oracle Communications Session Border Controller.</p> <p>21 - Oracle Communications Session Border Controller sends re-INVITE to SRS with new SDP and metadata.</p> <p>22 - SRS responds with OK to Oracle Communications Session Border Controller.</p> <p>23 - RTP stream initiated between Oracle Communications Session Border Controller and UA-B.</p> |
|---|--|

Call Flow Description

9 - RTP stream initiated between UA-A and Oracle Communications Session Border Controller.	24 - RTP stream initiated between Oracle Communications Session Border Controller and UA-C.
10 - RTP stream initiated between Oracle Communications Session Border Controller and UA-B.	25 - RTP stream initiated between Oracle Communications Session Border Controller and SRS.
11 - RTP stream initiated between Oracle Communications Session Border Controller and SRS.	26 - UA-C sends BYE to Oracle Communications Session Border Controller.
12 - UA-A sends REFER-TO: C to Oracle Communications Session Border Controller.	27 - Oracle Communications Session Border Controller responds with OK to UA-C.
13 - Oracle Communications Session Border ControllerOracle Communications Session Border Controller responds with 202 ACCEPTED to UA-A.	28 - Oracle Communications Session Border Controller sends BYE to UA-B.
14 - Oracle Communications Session Border Controller sends INVITE to UA-C.	29 - UA-B responds with OK to Oracle Communications Session Border Controller.
15 - UA-C responds with OK to Oracle Communications Session Border Controller.	30 - Oracle Communications Session Border Controller sends BYE to SRS.
	31 - SRS responds with OK to Oracle Communications Session Border Controller.

SRS Indicates Busy in Call (recording not required)

The following illustration shows the Session Recording Server (SRS) is BUSY for a call session. Recording is not required in this call flow.



Call Flow Description

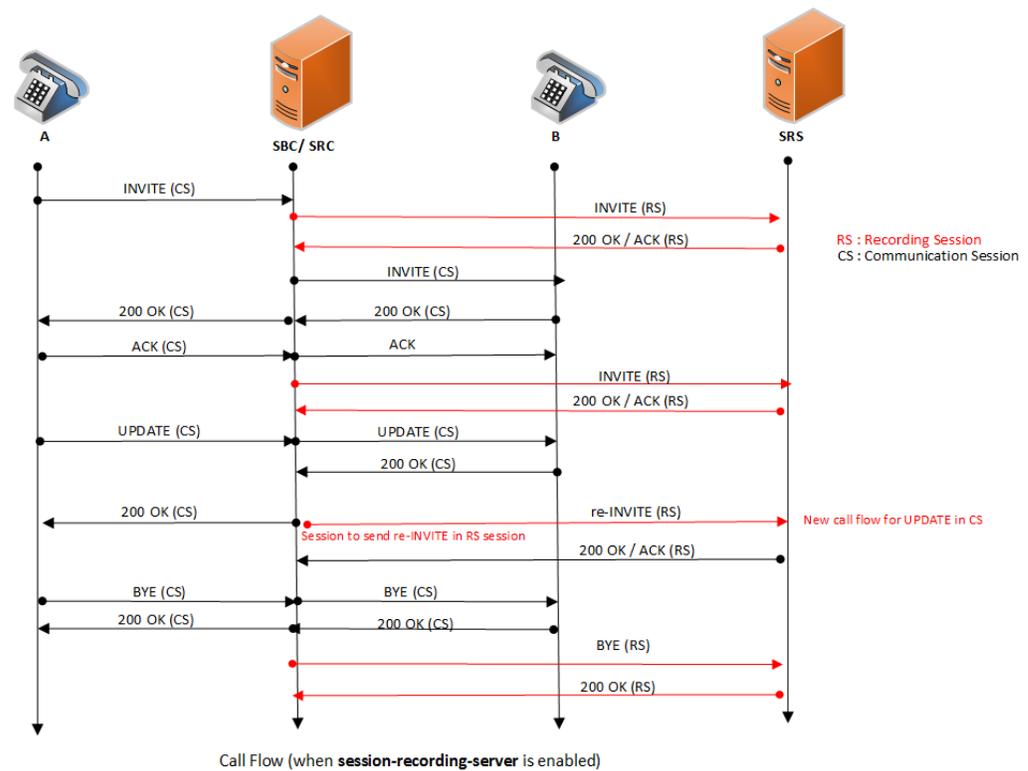
- ① UA-A sends INVITE to Oracle Communications Session Border Controller.
- ② Oracle Communications Session Border Controller forwards INVITE to UA-B.
- ③ UA-B responds with OK to Oracle Communications Session Border Controller.
- ④ Oracle Communications Session Border Controller forwards OK response to UA-A.
- ⑤ Oracle Communications Session Border Controller sends INVITE to SRS1 with SDP and metadata.
- ⑥ SRS1 responds to Oracle Communications Session Border Controller with 436 BUSY HERE.
- ⑦ RTP stream initiated between UA-A and Oracle Communications Session Border Controller.
- ⑧ RTP stream initiated between Oracle Communications Session Border Controller and UA-B.
- ⑨ Oracle Communications Session Border Controller sends INVITE to SRS2 with SDP and metadata.
- ⑩ SRS2 responds with OK to Oracle Communications Session Border Controller.
- ⑪ RTP stream initiated between Oracle Communications Session Border Controller and SRS2.
- ⑫ UA-A sends BYE to Oracle Communications Session Border Controller.
- ⑬ Oracle Communications Session Border Controller responds with OK to UA-A.
- ⑭ Oracle Communications Session Border Controller sends BYE to UA-B.
- ⑮ UA-B responds with OK to Oracle Communications Session Border Controller.
- ⑯ Oracle Communications Session Border Controller sends BYE to SRS2.

Call Flow Description

⑰ SRS2 responds with OK to Oracle Communications Session Border Controller.

Call Transfer Scenario (recording required)

The following illustration shows the Re-INVITE from Oracle Communications Session Border Controller to SRS on receiving in-dialog-requests INVITE/ UPDATE/ Re-INVITE during call session. Recording is required in this call flow.



Call Flow Description

1 - UA-A sends INVITE to Oracle Communications Session Border Controller.
 2 - Oracle Communications Session Border Controller updates the INVITE message to the Metadata in SRS.
 3 - SRS responds with OK to Oracle Communications Session Border Controller.
 4 - Oracle Communications Session Border Controller sends an INVITE to UA-B.
 5 - UA-B responds with OK to Oracle Communications Session Border Controller.

11 - UA-B responds with OK to Oracle Communications Session Border Controller.
 12 - Oracle Communications Session Border Controller forwards OK to UA-A..
 13 - Oracle Communications Session Border Controller sends a re-INVITE to SRS in Recording Session.
 14 - SRS responds with OK to Oracle Communications Session Border Controller.
 15 - UA-A sends BYE to Oracle Communications Session Border Controller.

Call Flow Description

6 - Oracle Communications Session Border Controller forwards OK response to UA-A.	16 - Oracle Communications Session Border Controller forwards BYE to UA-B.
7 - Oracle Communications Session Border Controller sends INVITE to SRS in Recording Session.	17 - UA-B responds with OK to Oracle Communications Session Border Controller.
8 - SRS responds with OK to Oracle Communications Session Border Controller.	18 - Oracle Communications Session Border Controller forwards OK to UA-B.
9 - UA-A sends an UPDATE message to Oracle Communications Session Border Controller.	19 - Oracle Communications Session Border Controller sends BYE to SRS in Recording Session.
10 - Oracle Communications Session Border Controller forwards UPDATE message to UA-B.	20 - SRS responds with OK to Oracle Communications Session Border Controller in Recording Session.

Oracle Communications Session Monitor Mediation Engine

The Oracle® Communications Session Monitor Mediation Engine is a platform that collects SIP, DIAMETER, DNS and ENUM protocol message traffic received from Communications Session Monitor Probes. A Probe is software run on COTS hardware; it is deployed within a network and collects packets from span/monitor ports on Ethernet switches, or receives IP-in-IP tunneled packet-traces from Oracle's Oracle Communications Session Border Controller. A Probe takes the protocol packets, prepends a receive timestamp and other information, encapsulates the packets, and passes them to the Communications Session Monitor via a secure connection. After receiving protocol traffic from a Probe, the Communications Session Monitor stores the traffic in an internal database, and analyzes aggregated data to provide comprehensive multi-level monitoring, troubleshooting, and interoperability information.

In contrast to the Packet-Trace feature, message logging is performed by software, which sends a copy of sent/received messages over UDP, or by saving such messages in a local file. The copy includes a timestamp, port/vlan information, and IP:port information, but all in ASCII format. Message Logging is performed after all decryption, meaning that SIP/TLS traffic can be monitored. Because remote message logging sends the protocol messages over UDP, there is no guarantee or confirmation of delivery.

The Oracle Communications Session Border Controller provides support for Communications Monitoring, a user-configurable capability that enables the system to function as a Communications Session Monitor Probe. Acting as a Probe, or as an exporter, the Oracle Communications Session Border Controller can:

1. Establish an authenticated, persistent, reliable TCP connection between itself and one or more Communications Session Monitor Mediation Engines.
2. Optionally ensure message privacy by encrypting the TCP connection using TLS.
3. Use the TCP connection to send a UTC-timestamped, unencrypted copy of a protocol message to the Communications Session Monitor Engine(s).
4. Accompany the copied message with related data to include: the port/vlan on which the message was sent/received, local and remote IP:port information, and the transport layer protocol.

IPFIX

The Oracle Communications Session Border Controller uses the IPFIX suite of standards to export protocol message traffic and related data to the Communications Session Monitor Mediation Engine.

- RFC 5101, *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information*
- RFC 5102, *Information Model for IP Flow Information Export*
- RFC 5470, *Architecture for IP Flow Information Export*
- RFC 5655, *Specification of the IP Flow Information Export (IPFIX) File Format*
- RFC 5815, *Definitions of Managed Objects for IP Flow Information Export*

The IPFIX standards describe the use of templates to format the export of specific types of protocol traffic. The Oracle Communications Session Border Controller and the Communications Session Monitor Mediation Engine share ten (10) pre-defined templates that facilitate protocol message exchange, and subsequent processing and analysis by the Communications Session Monitor Engine.

The pre-defined templates are:

- incoming SIP/DNS over UDP
- incoming SIP over TCP
- incoming SIP over SCTP
- incoming DNS over UDP (entire IP and UDP header not included)
- outgoing SIP/DNS over UDP
- outgoing SIP over TCP
- outgoing SIP over SCTP
- outgoing DNS over UDP (entire IP and UDP header not included)
- media qos and flow record
- IPFIX handshake (used for connection establishment)

Communications Monitor Configuration

Communications Monitor configuration consists of the following steps.

1. Configuration of one or more Oracle Communications Session Border Controller/Palladium exporter/collector pairs.

Configuration of the -config object, which defines common operations across all interfaces, is not required. Default values can be retained to enable standard service.

2. Optional assignment of a TLS profile to an exporter/collector pair.

Communication Monitor

Use the following procedure to configure communication monitoring:

1. From superuser mode, use the following ACLI sequence to access comm-monitor configuration mode. From comm-monitor mode, you establish a connection between the Oracle Communications Session Border Controller, acting as a exporter of protocol message traffic and related data, and an Oracle® Communications Session Monitor Mediation Engine, acting as an information collector.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# system
ACMEPACKET(system)# system-config
ACMEPACKET(system-config)# comm-monitor
ACMEPACKET(comm-monitor)#
```

2. Use the **state** parameter to enable or disable communication monitoring.

Communication monitoring is disabled by default.

```
ACMEPACKET(comm-monitor)# state enabled
ACMEPACKET(comm-monitor)#
```

3. Use the **sbc-group-id** parameter to assign an integer value to the Oracle Communications Session Border Controller, in its role as an information exporter.

Retain the default value (0) or assign another integer value.

```
ACMEPACKET(comm-monitor)# sbc-group-id 5
ACMEPACKET(comm-monitor)#
```

4. If the network interface specified in Step 8 is a media interface, you can optionally use TLS to encrypt the exporter/collector connection.

To enable TLS encryption, use the **tls-profile** parameter to identify a TLS profile to be assigned to the network interface. The absence of an assigned TLS profile (the default state) results in unencrypted transmission.

Refer to [TLS Profile Configuration](#) for configuration details.

```
ACMEPACKET(comm-monitor)# tls-profile commMonitor
ACMEPACKET(comm-monitor)#
```

5. Use the **qos-enable** parameter to enable or disable to export of RTP, SRTP, and QOS data flow information.

```
ACMEPACKET(comm-monitor)# qos-enable enabled
ACMEPACKET(comm-monitor)#
```

6. Use the **monitor-collector** parameter to move to monitor-collector configuration mode.

While in this mode you identify a Communications Session Monitor Mediation Engine collector by IP address and port number.

```
ACMEPACKET(comm-monitor)# monitor-collector
ACMEPACKET(monitor-collector)#
```

7. Use the **address** and **port** parameters to specify the IP address and port number monitored by a Communications Session Monitor Mediation Engine for incoming IPFIX traffic.

Enter an IPv4 address and a port number **with values either 4739 (unsecured) or 4740 (secured)**. The default value for the port is **4739**.

```
ACMEPACKET(monitor-collector)# address 172.30.101.239
ACMEPACKET(monitor-collector)# port 4739
ACMEPACKET(monitor-collector)#
```

8. Use the **network-interface** parameter to specify the network interface that supports the TCP connection between the Oracle Communications Session Border Controller to the Communications Session Monitor Mediation Engine.

To specify the wancom0 management interface:

```
ACMEPACKET(comm-monitor)# network-interface wancom0:0
ACMEPACKET(comm-monitor)#
```

To specify a media interface:

```
ACMEPACKET(comm-monitor)# network-interface m01
ACMEPACKET(comm-monitor)#
```



Note:

If configuring with a media interface, that interface must belong to a configured realm.

9. Use **done** and **exit** to return to comm-monitor configuration mode.
10. Use **done**, **exit**, and **verify-config** to complete configuration.
11. Repeat Steps 1 through 10 to configure additional as required.

TSCF Rekey Profile Configuration

Rekeying is a cryptographic technique that enhances security by enforcing the negotiation of existing keys on an ongoing secure connection. Rekeying can be either time-based, in which case new keys are negotiated at the expiration of a timer, or traffic-based, in which case new keys are negotiated when a threshold byte count is exceeded.

Use the following procedure to configure an optional tscf-rekey-profile. Later, you will assign the profile to a specific TSCF interface. If you do not intend to enforce re-keying, this procedure can be safely ignored.

1. From superuser mode, use the following command sequence to access tscf-rekey-profile configuration mode.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# security
ACMEPACKET(security)# tscf
ACMEPACKET(tscf)# tscf-rekey-profile
ACMEPACKET(tscf-rekey-profile)#
```

2. Use the **name** parameter to provide a unique identifier for this tscf-rekey-profile.

```
ACMEPACKET(tscf-rekey-profile)# name tscfRekey01
ACMEPACKET(tscf-rekey-profile)#
```

3. Use the **initiator** parameter to identify the rekey initiator.

Supported values are **client** (default) | **server** (the Session Director)

```
ACMEPACKET(tscf-rekey-profile)# initiator client
ACMEPACKET(tscf-rekey-profile)#
```

4. Use the **max-rekey-time** parameter to specify the maximum interval (in minutes) between re-keying operations.

Supported values are **0** (default) | **30 - 1440** (minutes)

The default value, **0**, specifies that time-based rekeying is not enforced; other integer values specify that time-based re-keying must be initiated by the tunnel endpoint designated by the **initiator** parameter.

```
ACMEPACKET(tscf-rekey-profile)# max-rekey-time 30
ACMEPACKET(tscf-rekey-profile)#
```

5. Use the **max-rekey-data** parameter to specify the maximum traffic exchange (measured in Kb) between rekeying operations.

The default value, **0**, specifies that traffic-based rekeying is not enforced; other integer values specify that traffic-based re-keying must be initiated by the tunnel endpoint designated by the **initiator** parameter.

```
ACMEPACKET(tscf-rekey-profile)# max-rekey-data 0
ACMEPACKET(tscf-rekey-profile)#
```

6. Use **done**, **exit**, and **verify-config** to complete tscf-rekey-profile configuration.
7. Repeat Steps 1 through 6 to configure additional tscf-rekey-profiles as required.

TLS Profile Configuration

Use the following procedure to configure a `tls-profile` that identifies the cryptographic resources, specifically certificates and protocols, required for the establishment of a secure/encrypted connection between the Oracle Communications Session Border Controller and the Communications Session Monitor Mediation Engine.

1. From superuser mode, use the following command sequence to access `tls-profile` configuration mode.

```
ACMEPACKET# configure terminal
ACMEPACKET(configure)# security
ACMEPACKET(security)# tls-profile
ACMEPACKET(tls-profile)#
```

2. Use the **name** parameter to provide a unique identifier for this `tls-profile`.

```
ACMEPACKET(tls-profile)# name commMonitor
ACMEPACKET(tls-profile)#
```

3. Use the required **end-entity-certificate** parameter to specify the name of the certificate-record configuration that identifies the credential (specifically, an X509.v3 certificate) offered by the Oracle Communications Session Border Controller in support of its asserted identity.

```
ACMEPACKET(tls-profile)# end-entity-certificate commMonitor509
ACMEPACKET(tls-profile)#
```

4. Use the required **trusted-ca-certificates** parameter to compile a list or one or more certificate-record configuration elements referencing trusted Certification Authority (CA) certificates used to authenticate the offered certificate. These referenced certificates are conveyed to the Communications Session Monitor Mediation Engine as part of the TLS exchange.

Provide a comma separated list of existing CA **certificate-record** configuration elements.

```
ACMEPACKET(tls-profile)# trusted-ca-certificates verisignClass3-
a,verisignClass3-b,baltimore,thawtePremium,acme-CA
ACMEPACKET(tls-profile)#
```

5. Retain the default value, **all**, for the **cipher-list** parameter.
6. Use the **verify-depth** parameter to specify the maximum number of chained certificates that will be processed while authenticating end-entity certificate received from the Communications Session Monitor Mediation Engine.

Provide an integer within the range 1 through 10 (the default).

The Oracle Communications Session Border Controller supports the processing of certificate chains (consisting of an end-entity certificate and some number of CA certificates) when X.509v3 certificate-based authentication is used. The following process validates a received TLS certificate chain.

- Check the validity dates (Not Before and Not After fields) of the end certificate. If either date is invalid, authentication fails; otherwise, continue chain validation
- Check the maximum length of the certificate chain (specified by `verify-depth`). If the current chain exceeds this value, authentication fails; otherwise, continue chain validation.
- Verify that the Issuer field of the current certificate is identical to the Subject field of the next certificate in the chain. If values are not identical, authentication fails; otherwise, continue chain validation.
- Check the validity dates (Not Before and Not After fields) of the next certificate. If either date is invalid, authentication fails; otherwise, continue chain validation.
- Check the X509v3 Extensions field to verify that the current certificate identifies a CA. If not so, authentication fails; otherwise, continue chain validation.
- Extract the Public Key from the current CA certificate. Use it to decode the Signature field of the prior certificate in the chain. The decoded Signature field yields an MD5 hash value for the contents of that certificate (minus the Signature field).
- Compute the same MD5 hash. If the results are not identical, authentication fails; otherwise, continue chain validation.
- If the hashes are identical, determine if the CA identified by the current certificate is a trust anchor by referring to the `trusted-ca-certificates` attribute of the associated TLS-profile configuration object. If the CA is trusted, authentication succeeds. If not, return to Step 2.

```
ACMEPACKET(tls-profile)# verify-depth 8
ACMEPACKET(tls-profile)#
```

7. Use the **mutual-authenticate** parameter to **enable** or **disable** (the default) mutual authentication.

Protocol requirements mandate that the server present its certificate to the client application. Optionally, the server can implement mutual authentication by requesting a certificate from the client application, and authenticating the certificate offered by the client.

Upon receiving a server certificate request, the client application must respond with a certificate; failure to do so results in authentication failure.

```
ACMEPACKET(tls-profile)# mutual-authenticate disabled
ACMEPACKET(tls-profile)#
```

8. Retain the default value, **compatibility**, for the **tls-version** parameter.
9. Retain default values for all other parameters.
10. Use **done**, **exit**, and **verify-config** to complete `tls-profile` configuration.
11. Repeat Steps 1 through 10 to configure additional `tls-profiles` as required.

Oracle Communications Session Monitor Statistics

The Oracle Communications Session Border Controller collects statistics on the operations of its communications monitor probe, which provides protocol traffic information to Oracle

Communications' Session Monitor. The user displays information about the connections to Session Monitor servers using the **show comm-monitor** command. The user can set all comm-monitor statistics to zero using the command **reset comm-monitor**.

This Command shows three types of aggregate statistics for all clients, including:

- Client State
- Socket Statistics
- Other Aggregate Statistics

The Oracle Communications Session Border Controller's **show comm-monitor** command shows the Lifetime statistics. The command runs without or with arguments, including:

- **show comm-monitor**—Shows client connection states and aggregate stats
- **show comm-monitor by-client <IP-Addr>**—Shows client stats
- **show comm-monitor errors**—Shows Errors captured by the Session Monitor
- **show comm-monitor internal**—Shows Oracle Communications Session Border Controller traffic-specific counters

Example command output without arguments is presented below:

```
ORACLE# show comm-monitor
Client                               State          PROTOCOL
=====
192.168.42.10:4739                   Out-of-Service TCP
192.168.42.11:4739                   Connecting     TCP
192.168.42.12:4739                   In-Service    TCP

15:06:10-35 (recent)
CommMonitor Socket Statistics
-----
Recent          Total          PerMax
-----
Socket Message Dropped      0              0          0
Socket Send Error           0              0          0
Socket Not Ready            0              0          0
Socket Timeouts             0              0          0
Socket Disconnects         0              0          0
Socket Reconnects          0              0          0
Buffer Allocation Error     0              0          0

CommMonitor Statistics
-----
Recent          Total          PerMax
-----
Handshake Msg Sent          0              0          0
Handshake Msg ACK           0              0          0
Handshake Msg NAK           0              0          0
Keep Alive                  0              0          0
SIP UDP Recv Msg Sent       0              0          0
SIP UDP Send Msg Sent       0              0          0
SIP TCP Recv Msg Sent       0              0          0
SIP TCP Send Msg Sent       0              0          0
SIP SCTP Recv Msg Sent      0              0          0
SIP SCTP Send Msg Sent      0              0          0
ENUM Recv Msg Sent          0              0          0
ENUM Send Msg Sent          0              0          0
```

Packet Trace

The Oracle Communications Session Border Controller's packet trace tool provides the user with the ability to capture traffic from the Oracle Communications Session Border Controller itself.

The user invokes the tool from the ACLI, manually specifying:

- How to capture (local vs remote)
- What to capture
- Capture start and stop

There are two capture modes, one that saves traffic locally and one that mirrors traffic to a user-specified target. Software only deployments support local capture only. Proprietary Acme hardware deployments support both local and remote capture.

- Local capture supports PCAP filters to specify the type of traffic to capture. Remote capture supports its own syntax to identify the traffic to mirror.
- Local packet capture is dependent on access control configuration, not capturing any denied traffic. Remote capture mirrors traffic regardless of access control configuration.
- The system does not capture RTP via local packet capture.

Installed NIUs impact remote packet capture. Fragmented packets that ingress HIFNs or Cavium NIUs include only the outer header within the fragments. As a result, these traces do not appear to be using IP/IP encapsulation. This differs from fragmented packets that ingress the Quad port GiGE and Copper PHY NIUs. These traces include inner and outer headers within the fragments.

Do not run packet-trace simultaneously with other Oracle Communications Session Border Controller replication features, such as LI, SRS, SIP Monitoring and Trace, and Call Recording. These features may interfere with each other, corrupting each's results.

Packet Trace Remote

Packet trace remote enables the Oracle Communications Session Border Controller to mirror traffic between two endpoints, or between itself and a specific endpoint to a user-specified target. To accomplish this, the Oracle Communications Session Border Controller replicates the packets sent and received, encapsulates them according to RFC 2003, and sends them to a user-configured target. At the target, the user would capture and analyze the packets.

Currently, the Oracle Communications Session Border Controller supports:

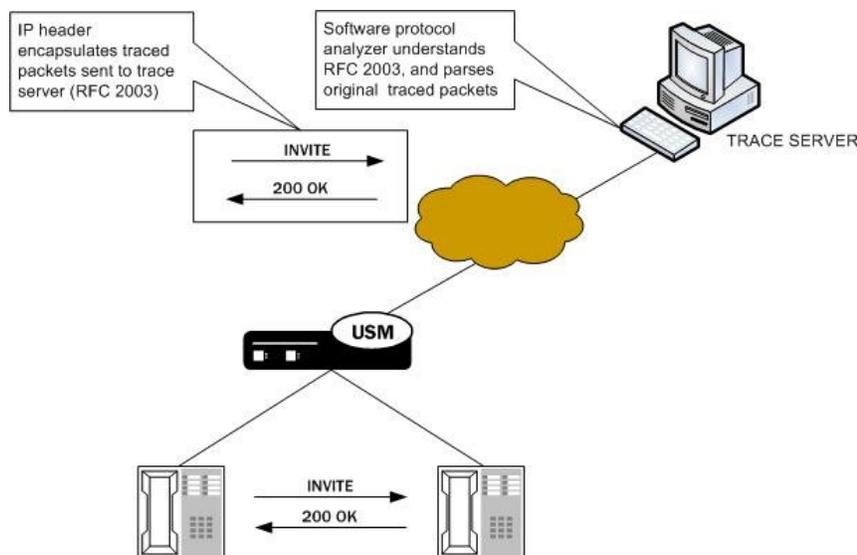
- One configurable trace server (on which you capture/analyze the traffic)
- Sixteen concurrent endpoint traces

To use this feature, the user configures a **capture-receiver** on the Oracle Communications Session Border Controller so that it knows where to send the mirrored packets. Once the **capture-receiver** is configured, the user issues the **packet-trace** command to start, stop and specify filters for traces.

You establish a packet trace filter with the following information:

- Network interface—The name of the network interface on the Oracle Communications Session Border Controller from which you want to trace packets. The user can enter this value as a name or as a name and subport identifier value (name:subportid)
- IP address—IP address of the endpoint to or from which the target traffic goes.
- Local port number—Optional parameter; Layer 4 port number on which the Oracle Communications Session Border Controller receives and from which it sends; if no port is specified or if it is set to 0, then all ports will be traced
- Remote port number—Optional parameter; Layer 4 port number to which the Oracle Communications Session Border Controller sends and from which it receives; if no port is specified or if it is set to 0, then all ports will be traced.

The Oracle Communications Session Border Controller then encapsulates the original packets in accordance with RFC 2003 (*IP Encapsulation within IP*); it adds the requisite headers, and the payload contains the original packet trace with the Layer 2 header removed. Since software protocol analyzers understand RFC 2003, they can easily parse the original traced packets.



It is possible that—for large frames—when the Oracle Communications Session Border Controller performs the steps to comply with RFC 2003 by adding the requisite header, the resulting packet might exceed Ethernet maximum transmission unit (MTU). This could result in packets being dropped by external network devices, but widespread support for jumbo frames should mitigate this possibility.

If the Oracle Communications Session Border Controller either receives or transmits IP fragments during a packet trace, it only traces the first fragment. The first fragment is likely to be a maximum-sized Ethernet frame.

The Oracle Communications Session Border Controller continues to conduct the packet trace and send the replicated information to the trace server until you instruct it to stop. You stop a packet trace with the ACLI **packet-trace remote stop** command. With this command, you can stop either an individual packet trace or all packet traces that the Oracle Communications Session Border Controller is currently conducting.

Packet Trace Local

Packet Trace Local enables the Oracle Communications Session Border Controller to capture traffic between two endpoints, or between itself and a specific endpoint. To accomplish this, the

Oracle Communications Session Border Controller replicates the packets sent and received and saves them to disk in .PCAP format.

The default packet trace filter uses the specified interface to capture both ingress and egress traffic. To specify captured traffic, the user can append the command with a PCAP filter enclosed in quotes. PCAP filter syntax is widely published.

While capturing, the system displays the number of packets it has captured and prevents the user from entering any other ACLI commands from that session. The user terminates the capture by pressing Ctrl+C.

By default, the system saves the .PCAP file in /opt/traces, naming it with the applicable interface name as well as the date and time of the capture. Alternatively, the user can specify file name using the system supports the PCAP filter flags -w.

The system rotates the PCAPs created in this directory by size. The last 25 files are kept and are rotated when they reach 100 MB. If there are capture files in the /opt/traces directory when this command is run, the system prompts the user to remove them before running new captures. If preferred, the user can decline this file deletion.

Packet Trace Scenarios

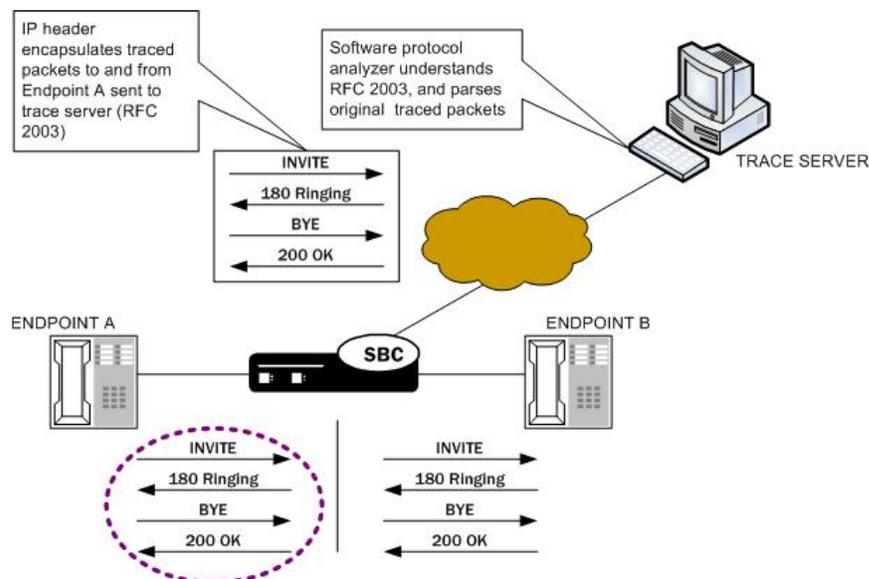
This section describes three possible ways that you might use the packet trace feature. You can examine communications sent to and from one endpoint, sent between two endpoints, or sent between ingress and/or egress Oracle Communications Session Border Controller interfaces to endpoints.

Packet Trace for One Endpoint

When you use the `packet-trace remote <state>` command, the Oracle Communications Session Border Controller sets up packet tracing for one endpoint. The Oracle Communications Session Border Controller collects and replicates the packets to and from one endpoint. To enable this kind of trace, you set up one packet trace using the **packet-trace** command.

The commands you carry out for **packet-trace remote** would take the following form:

```
ORACLE# packet-trace remote start F01:0 <IP address of Endpoint A>
```



The commands you carry out for **packet-trace local** would take the following form:

```
ORACLE# packet-trace local F01:0 <"host IP address of Endpoint A">
```

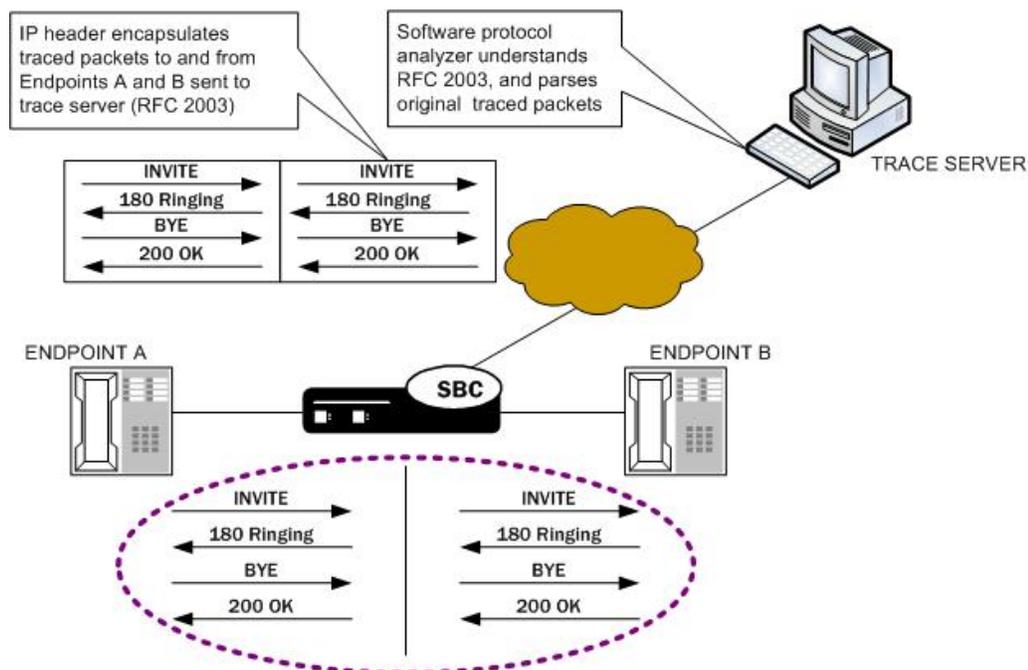
Packet Trace for Both Call Legs

If you want to trace both sides (both call legs), then you must set up individual traces for each endpoint—meaning that you would initiate two packet traces. The results of the trace will give you the communications both call legs for the communication exchanged between the endpoints you specify.

If you initiate a packet trace for both endpoints that captures both signaling and media, the signaling will be captured as usual. However, RTP will only be traced for the ingress call leg. This is because the Oracle Communications Session Border Controller performs NAT on the RTP, which means it cannot be captured on the egress call leg.

The commands you carry out for **packet-trace remote** would take the following form:

```
ORACLE# packet-trace remote start F01:0 <IP address of Endpoint A>
ORACLE# packet-trace remote start F02:0 <IP address of Endpoint B>
```



The commands you carry out for **packet-trace local** would take the following form:

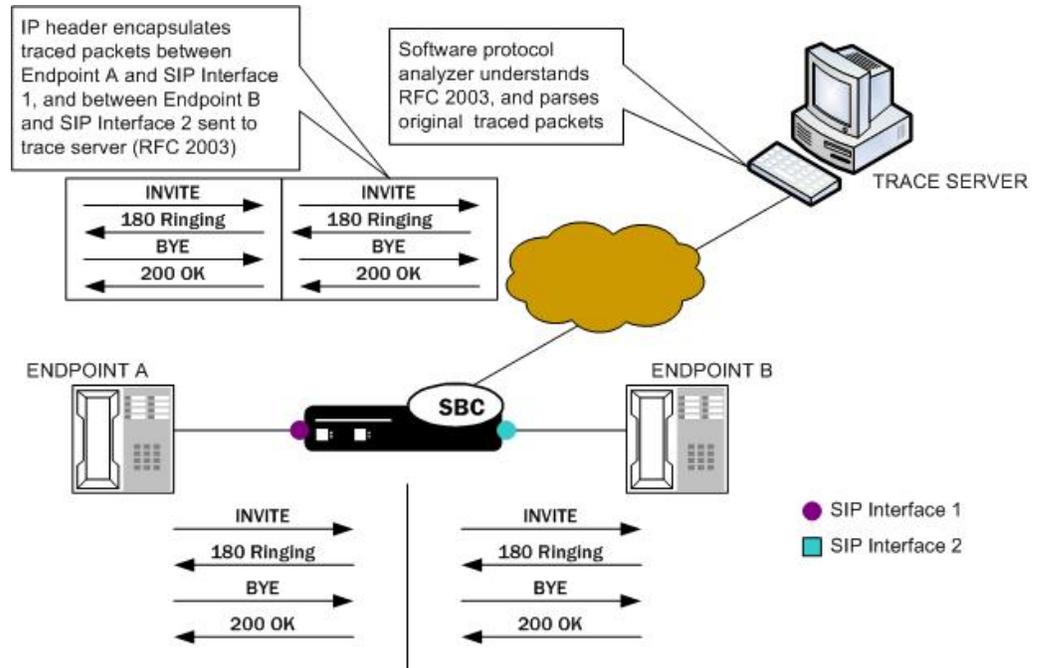
```
ORACLE# packet-trace local F01:0 <"host IP address of Endpoint A">
ORACLE# packet-trace local F02:0 <"host IP address of Endpoint B">
```

Packet Trace for a Oracle Communications Session Border Controller Signaling Address

You can perform a packet trace for addresses internal to the Oracle Communications Session Border Controller; this can be the address, for example, of a SIP interface. Using signaling interface addresses puts the emphasis on the Oracle Communications Session Border Controller rather than on the endpoints by allowing you to view traffic from specified interfaces.

The commands you carry out for **packet-trace remote** would take the following form:

```
ORACLE# packet-trace remote start F01:0 <IP address of Oracle Communications
Session Border Controller interface1>
ORACLE# packet-trace remote start F02:0 <IP address of Oracle Communications
Session Border Controller interface2>
```



The commands you carry out for **packet-trace local** would take the following form:

```
ORACLE# packet-trace local F01:0 <"host IP address of Oracle Communications
Session Border Controller interface1">
ORACLE# packet-trace local F02:0 <"host IP address of Oracle Communications
Session Border Controller interface2">
```

Note:

The system does not support egress RTP capture with Transcoding NIU

Running Packet Trace

There are four steps you take when you use the **packet-trace remote** feature. For **packet-trace local**, there are only two.

- Configuring the Oracle Communications Session Border Controller with the trace server information so that the Oracle Communications Session Border Controller knows where to send replicated data. (**packet-trace remote** only)
- Setting up the capture filter `ip proto 4` in your software protocol analyzer if you only want to see the results of the Oracle Communications Session Border Controller packet trace(s). (**packet-trace remote** only)
- Starting a packet trace.

- Stopping a packet trace.

This section provides information about how to perform all tasks.

Configuring a Trace Server

Trace servers only apply to **packet-trace remote**. You need to configure a trace server on the Oracle Communications Session Border Controller; this is the device to which the Oracle Communications Session Border Controller sends replicated data. The Oracle Communications Session Border Controller supports one trace server.

To configure a trace server on your Oracle Communications Session Border Controller:

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

```
ACMEPACKET# configure terminal
```

2. Type **system** and press Enter.

```
ACMEPACKET(configure)# system  
ACMEPACKET(system)#
```

3. Enter **capture-receiver** and press Enter.

```
ACMEPACKET(system)# capture-receiver  
ACMEPACKET(capture receiver)#
```

4. **state**—Type **enabled** so that you can use the trace server to which you want to send the mirrored packets for calls you are packet tracing. The default is **disabled**. The valid values are:

- enabled | disabled

Disable capture receivers you are not actively using for traces to prevent potential service outages caused by the capture's system resource utilization.

5. **address**—Enter the IP address of the trace server; there is no default.
6. **network-interface**—Enter the name and subport of the Oracle Communications Session Border Controller network interface from which the Oracle Communications Session Border Controller is to send mirrored packets. Your entry needs to take the form name:subport. The default is :0.
7. Save and activate your configuration.

Starting a Remote Packet Trace

You use the start a remote packet trace by entering the appropriate ACLI command with these pieces of information:

- Network interface (name:subport ID combination)
- IP address to be traced; if you do not enter local and/or remote ports when you start the trace, the Oracle Communications Session Border Controller will trace all ports
- (Optional) Local UDP/TCP port on which the Oracle Communications Session Border Controller sends and receives traffic to be traced
- (Optional) Remote UDP/TCP port to which the Oracle Communications Session Border Controller sends traffic, and from which it receives traffic to be traced; you cannot enter the remote port without specifying a local port

To start a packet trace with local and remote ports specified:

Note that the system supports packet trace on the Acme Packet 3820 and Acme Packet 4500, but not on the Acme Packet 1100 and VM Edition.

1. Enter the ACLI **packet-trace remote** command followed by a Space, and the word **start**. After another Space, type in the name and subport ID for the network interface followed by a Space, the IP address to be traced followed by a Space, the local port number followed by a Space, and then optionally the remote port number. Then press Enter.

```
ACMEPACKET# packet-trace remote start core:0 192.168.10.99 5060 5060
Trace started for 192.168.10.99
```

Stopping a Remote Packet Trace

You stop a remote packet trace by entering the appropriate ACLI command with these pieces of information:

- Network interface (name:subport ID combination)
- IP address to be traced
- (Optional) Local UDP/TCP port on which the Oracle Communications Session Border Controller sends and receives traffic to be traced
- (Optional) Remote UDP/TCP port to which the Oracle Communications Session Border Controller sends traffic, and from which it receives traffic to be traced

If the packet trace you want to stop has no entries for local and/or remote ports, then you do not have to specify them.

1. To stop a packet trace with local and remote ports specified, enter the ACLI **packet-trace remote** command followed by a Space, and the word **stop**. After another Space, type in the name and subport ID for the network interface followed by a Space, the IP address to be traced followed by a Space, the local port number followed by a Space, and then optionally the remote port number. Then press Enter.

```
ACMEPACKET# packet-trace remote stop core:0 192.168.10.99 5060 5060
```

2. To stop all packet traces on the Oracle Communications Session Border Controller, enter the ACLI **packet-trace remote** command followed by a Space, and the word **stop**. After another Space, type the word **all** and press Enter.

```
ACMEPACKET# packet-trace remote stop all
```

Starting a Local Packet Trace

You use the start a packet trace by entering the appropriate ACLI command with these pieces of information:

- Network interface (name:subport ID combination)
- (Optional) Enter a `tcpdump` command line within quotes

Note that the system supports local packet trace on all platforms.

1. Enter the ACLI **packet-trace local** command followed by a Space. Type in the name and subport ID for the network interface followed by a Space, the IP address to be traced followed by a Space, the local port number followed by a Space, and then optionally the remote port number. Then press Enter.

```
ACMEPACKET# packet-trace local s0p0 "host 192.168.12.12"
Files found in trace directory. Remove [y/n]?: y
```

```
File: /opt/traces/s0p0_0_00001_20150723095442.pcap  
Packets: 5 Packets dropped: 0
```

The ACLI session does not accept use input while the **packet-trace local** command is running.

Stopping a Local Packet Trace

Type Ctrl-C to stop a local packet trace. This also re-enables the command line session.

2

Web-based GUI

Introduction

This chapter provides information and procedures for using the embedded Web-based GUI on your Oracle Communications Session Border Controller. This GUI displays results of filtered SIP session data from one or multiple Oracle Communications Session Border Controllers, and provides traces in a common log format for local viewing or for exporting to your PC. The Web-based GUI client provides local viewing support of SIP sessions only.

Web Browser Support

You can use any of the following Web browsers to access Oracle Communications Web-based GUI for viewing SIP session data:

- Internet Explorer versions 9.0 and higher
- Mozilla Firefox versions 12.0 and higher
- Google Chrome versions 19.0.1084.46m and higher

Overview

The Web-based GUI allows you to view information about monitored SIP sessions on the Oracle Communications Session Border Controller(s). It supports four types of summary reports:

- **Sessions**
- **Registrations**
- **Subscriptions**
- **Notable events (interesting events)**

Each report summarizes the applicable information, displays it on easy-to-read pages, and allows you to sort and customize columns within each report. You can also perform searches to find specific information in a report.

Each type of report provides sorting, searching, and paging functionality and provides action buttons to use to display additional information or perform a task.

Logging in

You can login to the Web-based GUI using your Web browser for browsers that support this GUI). The Web interface supports authentication functionality similar to a user logging in via TELNET, Secure Shell (SSH), File Transfer Protocol (FTP), and SSH File Transfer Protocol (SFTP). SIP Monitoring and Tracing also supports RADIUS authentication.

 **Note:**

Regardless of the privilege level set via authentication (user or admin), logging into the Web-based GUI gives user-level access privileges.

Up to 5 users can log into the same Web-based GUI interface. If more than 5 users attempt to log in, the following error message displays:

User limit reached. Please try again later.

To login to the Web-based GUI:

1. On your PC, open an Internet Browser.
2. Start the Web-based GUI by using either the HTTP or HTTPS login:

http://<Server IP address>
https://<Server IP address>

 **Note:**

Logging in using HTTP and/or HTTPS is dependant on the setting that was made during the Web-based GUI installation wizard. You can change these settings using the ACLI. For more information, see Configuring the Web Server.

3. Enter your Web-based GUI username and password. Default username is “**user**” or “**admin**”. Default password is packet.
4. Click <Login>.



5. The Web-based GUI opens.

Start Time	State	Call ID	Request URI	From URI	To URI	Ingress Reason	Egress Reason
2012-06-14 15:46:34.915	TERMINATED-200	5-17902@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:46:33.914	TERMINATED-200	4-17902@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:46:32.914	TERMINATED-200	3-17902@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:46:31.914	TERMINATED-200	2-17902@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:46:30.914	TERMINATED-200	1-17902@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:35.957	FAILED-406	5-17609@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:34.957	FAILED-406	4-17609@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:33.958	FAILED-406	3-17609@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:32.959	FAILED-406	2-17609@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:31.959	TERMINATED-0	1-17609@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:14.210	TERMINATED-200	5-17548@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:13.211	TERMINATED-200	4-17548@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core
2012-06-14 15:45:12.210	TERMINATED-200	3-17548@192.168.200.226	sip:teal@192.168.204.71:5060	*2273630@-tel:781-414-23...	out-sip:kam@192.168.204.7...	access	core

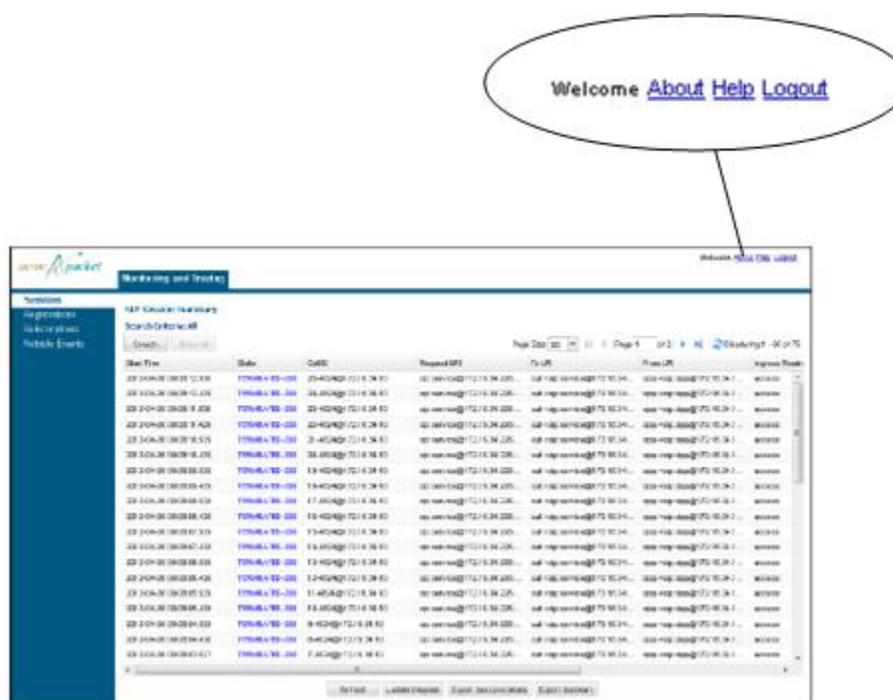
6. Click on the report you want to view in the slider on the left side of the page. On all Sessions, Registrations, Subscriptions, and Notable Events report pages, you can perform the following:
 - Customize the page display
 - Perform a search for specific data
 - Display a ladder diagram
 - Export session details
 - Export session summary
 The following paragraphs describe each of these tasks.
7. To logout of the Web-based GUI, click **Logout** in the upper right corner of the page.

About Information

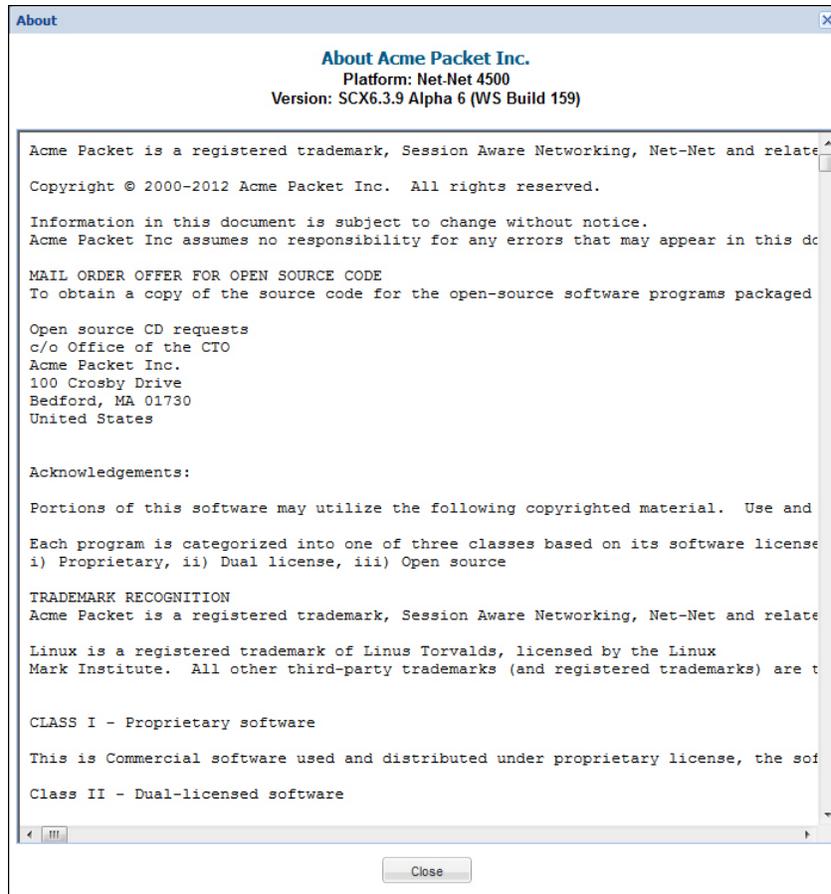
You can display information about the Oracle Communications Session Border Controller you are currently logged into via the Web-based GUI. You can access About from any main report summary page.

To display the About information:

1. From any main report summary page, click the About link in the upper right corner on the page.



The following information displays.



This box displays the following about the Oracle Communications Session Border Controller you are currently logged into:

- Acme Packet Platform type (3800, 4500, 4600, Server Edition, or Virtual Machine Edition)
 - Version Number of software (including Build Number)
 - Trademark Recognition
 - Copyright
 - Open Source Mailing Address
 - Acknowledgements
 - Relevant Proprietary and Open Source Licenses
2. Click <Close> to close the box.

Customizing the Page Display

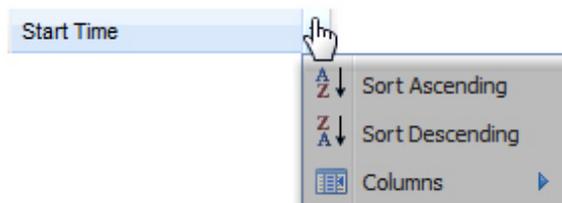
You can customize the data presented in the SM& T pages by changing whether or not specific columns display and how they display. You can also sort the order of item entries.

To customize the page display within any report:

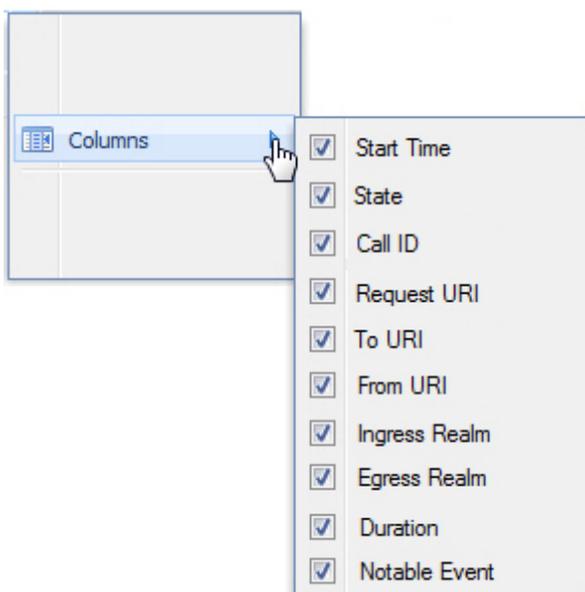
1. Position the cursor over a column heading. A pointer displays on the right hand side of the box. For example:



2. Click the down arrow to display the menu. For example:



3. Click Sort Ascending to sort the data in the table in ascending order.
4. Click Sort Descending to sort the data in the table in descending order.
5. Click Columns to access and customize a list of column names. For example:



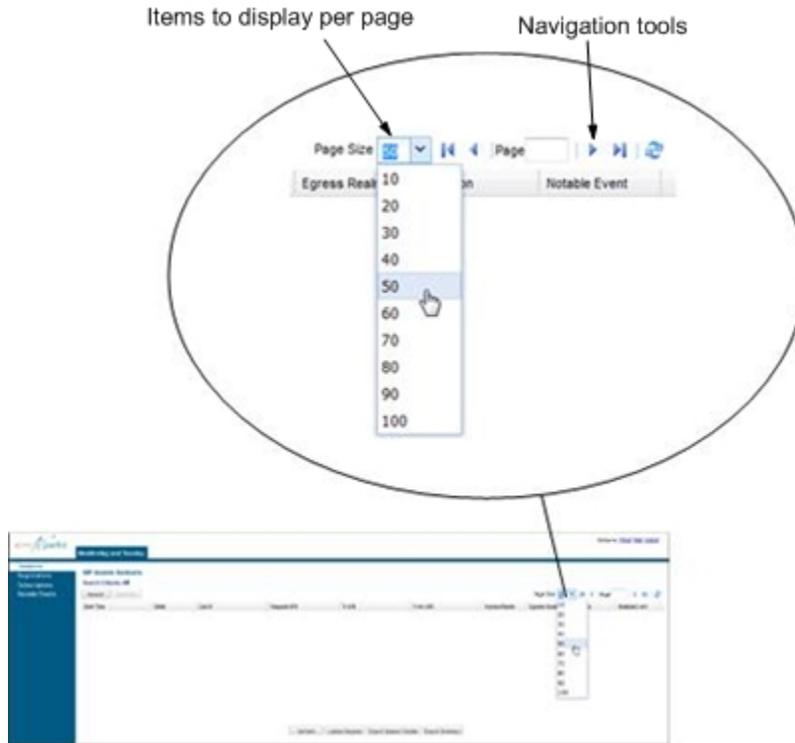
6. Place a checkmark in the box to display that heading/column in the window. Remove the checkmark to hide the heading/column in the window.
7. Click Group By This Field to group all items in the list according to the column head you currently have selected. For example, if you select the “Site” column heading, and then select Group By This Field, all items in the list are grouped by Site.
8. For the “**Show in Groups**” option, place a checkmark in the box to show the current heading in the group when you select Group By This Field. Remove the checkmark to hide the current heading when Group By This Field is selected.
9. To close the heading menus, click anywhere in the window.

Changing Number of Data Items on the Page

By default, 50 data items are shown per page. You can change the number of items viewed for all tables in Trunk Manager.

To change the number of data items displayed:

1. At the top right corner of the window, click the down arrow next to **Size**. The drop down list of values appears.

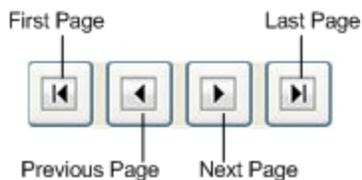


2. Click the number of data items you want to display per page. Default is **50**. Valid values are **10** to **100** in increments of 10.

Navigating Pages

To navigate through multiple pages:

1. Use the navigation arrows located at the top right corner of the window to navigate through multiple pages.
2. Click the navigation icons to display the desired page, such as the first page, previous page, next page, and the last page of Events list view.



Refresh

To refresh the page:

1. Click the <Refresh> button at the bottom of the page.

or

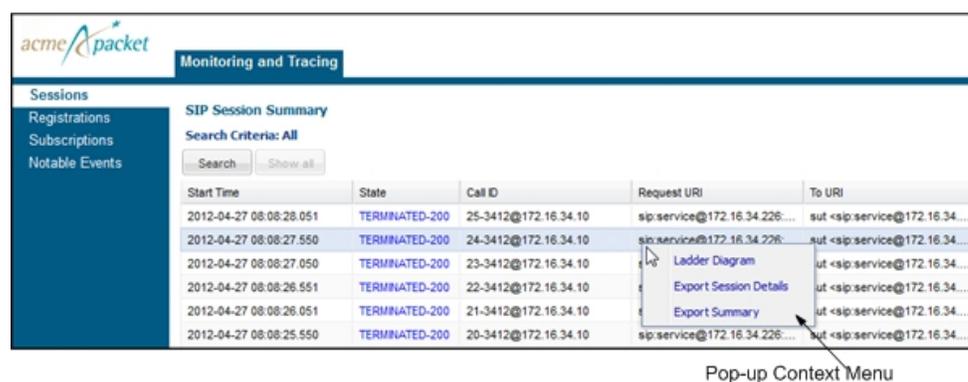
Click the  in the upper right corner of the page.

Pop-up Context Menu

All reports (Sessions, Registrations, Subscriptions, and Notable Events) support a pop-up context menu that you can use to select a Ladder diagram, Export Session Details, and Export Session Summary. You display the pop-up context menu by selecting a record and right-clicking to show a menu.

To display the pop-up context menu in any report:

1. Click on a record in the report page.



2. Right-click the mouse to display the pop-up context menu.
3. Drag the mouse to an option on the menu and left-click the mouse to select it. The applicable page displays or the export begins, depending on the option you selected (Ladder Diagram, Export Session Details, or Export Summary).

Help Menu

The Web-based GUI provides online help for all of the pages within the GUI. You can access help from any page by clicking the Help link in the upper right corner of the page.

To display Help:

1. From any main report summary page, click the Help link in the upper right corner on the page.

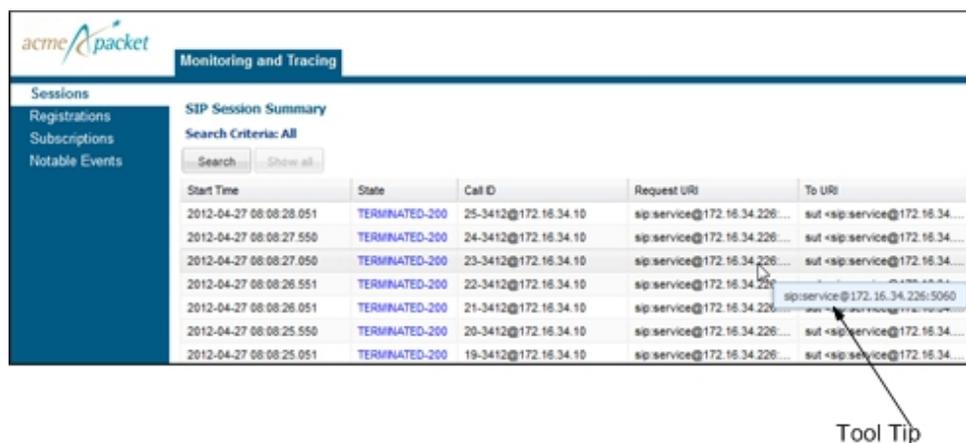
Click the X in the upper right corner of the pop-up window.

Tool-Tip Help

Throughout the Web-based GUI, you can scroll over a record within a summary report and display quick information about that record in a temporary pop-up box. This allows you to view a summarized version of help for that record.

To view a tool-tip help:

1. From any main report summary page, use the mouse to scroll over a record in a summary report.



A blue-shaded box displays allowing you to view quick help about the record.

2. To close the tool-tip, scroll off of the record. Alternatively, click at another location within the summary report.

Note:

Ladder diagrams for a specific record also display tool tip summary information. For more information, see [Ladder Diagram](#).

Search for a Record

The **<Search>** button at the top of the report page allows you to perform a search to find a specific record(s) within a report (Sessions, Registrations, Subscriptions, Notable Events). It also allows you to specify criteria on which to perform the search.

After defining a search criteria in the Search Filter dialog box, clicking **<Search>**, automatically populates the report page with the records that match the criteria you specified. The search performs the filtering process of criteria dependent on the report page from which you are running the search.

For example, performing a search from the Sessions report page displays only the reports pertaining to call sessions. If you performed a search on the Registration report page, only the reports pertaining to call registrations displays on the report page. The search string containing the criteria on which you performed the search, displays in the top left corner of the page.

Search String

The screenshot shows the 'acme packet' logo and a 'Monitoring and Tracing' section. Under 'Sessions', there is a 'SIP Session Summary' table. Above the table, search criteria are listed: 'Search Criteria: toURI=172.16.34.226; startDateTime=2012-04-26 08:08:28; endDateTime=2012-04-27 08:08:16'. Below this, there are 'Search' and 'Show all' buttons. The table has columns for Start Time, State, Call ID, Request URI, and To URI. An arrow points from the 'Search String' label to the search criteria text.

Start Time	State	Call ID	Request URI	To URI
2012-04-27 08:08:28:051	TERMINATED-200	25-3412@172.16.34.10	sip:service@172.16.34.226...	sut <sip:service@172.16...
2012-04-27 08:08:27:550	TERMINATED-200	24-3412@172.16.34.10	sip:service@172.16.34.226...	sut <sip:service@172.16...
2012-04-27 08:08:27:050	TERMINATED-200	23-3412@172.16.34.10	sip:service@172.16.34.226...	sut <sip:service@172.16...

The search criteria is saved until you click <Reset> in the dialog box, or until you log out of the HTTP session.

To perform a search:

Note:

You can specify a value for any or all of the fields in the Search box. The search process searches for records with all of the values you specify and displays only the records with these values. If you perform a “Global Search”, AND specify values in other fields, the search process searches the other specified fields first and then filters on the “Global Search” field.

Note:

If you specify a “*” in a search string, the search is performed on that exact string. For example, if you search for “123*45”, the search shows results for all strings containing “123*45”.

Note:

You can use quotes (“”) to specify a search. For example, you can enter Smith and the search finds all of the records that match Smith, such as:

```
John Smith field<sip:sipp@192.168.1.70:5070>;tag=12260SIPpTag001
```

If you enter a space before or after a quotation mark, (for example, "Smith "), the search returns no data.

1. In any reports page, click <Search>.
2. The following dialog box displays.

3. In the Global Search field, specify a string to search all parameters in all records. Valid values are alpha-numeric characters.

 **Note:**

The Global Search option searches all parameters in all the session records stored in memory. All values you specify in other fields are searched before the value specified in the Global Search field is used.

4. In the From URI field, enter the URI formatted string of the call source information you are searching. Valid values are alpha-numeric characters. For example, `sipp<sip:sipp@172.16.34.10:5060;tag=24`.
5. In the Requested URI field, enter the URI formatted string that contains a protocol, name, location, or any other applicable characteristic, that is sent by the Oracle Communications Session Border Controller in the REQUEST header. Valid values are alpha-numeric characters. For example, `sip:service@172.16.34.226:5060`.
6. In the To URI field, enter URI formatted string of the call destination information you are searching. Valid values are alpha-numeric characters. For example, `sut<sip:service@172.16.34.226:5060;tag=99`.

7. In the Start Date/Time (HH mm ss) field, enter a starting date to search on in the first text box in the format YYYY-MM-DD (where Y =year, M=month, and D=day). Alternatively, click on the calendar icon in this field to display a calendar from which you can select a date. Navigate the calendar to find the date you want and click on it to enter it into this field, or click <Today> to enter today's date. For example, 2012-04-15 would search for all records starting on April 15, 2012. Valid values are numeric characters only. Enter a start time to search on in the last three text boxes in the format HH mm ss (where H=hour, m=minutes, and s=seconds). For example, 01 30 45 would search for all records starting at 1:30 and 45 seconds. Valid value are numeric characters only.

Start Date/Time(HH mm ss)

End Date/Time(HH mm ss)



8. In the End Date/Time (HH mm ss) field, repeat the process of entering a date and time as provided in Step 7.
9. To search on additional parameters, click on the Additional Identifiers down arrow to expand the dialog box.

Additional Identifiers

Session ID

In Call ID

Out Call ID

State(with result code)

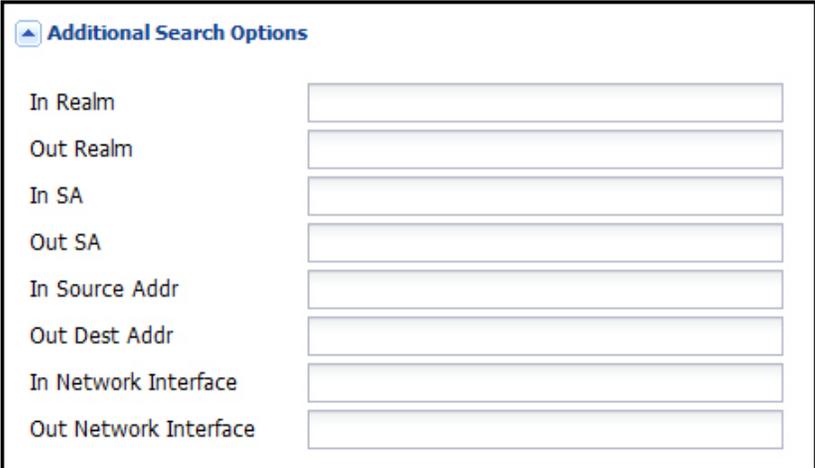
Notable Event

10. In the Session Id field, enter the ID of the call session you want to search. Valid values are alpha-numeric characters. For example, 22-3412@172.16.34.1.
11. In the In Call ID field, enter the ID of the incoming call (phone number and source IP address). Valid values are alpha-numeric characters. For example, 25-3412@172.16.34.10.
12. In the Out Call ID field, enter the ID of the outgoing call (phone number and IP address). Valid values are alpha-numeric characters. For example, 14-3412@172.14.54.6.
13. In the State (with result code) field, enter the status of the call session with the result code for which you want to search. Valid values are (case-sensitive):
 - INITIAL-<result code>

- EARLY-<result code>
- ESTABLISHED-<result code>
- TERMINATED-<result code>
- FAILED-<result code>

Result codes can range from 1xx to 5xx. For example, terminated-200, or failed-400.

14. In the Notable Event field, select the notable event for which you want to search. Valid values are:
 - any-event - search displays any notable event that was stored in memory.
 - short-session - search displays only records that indicate a short-session duration has occurred.
 - local-rejection - search displays only records that indicate a local-rejection has occurred.
15. To search on additional parameters, click on the Additional Search Options down arrow to expand the dialog box.



Additional Search Options	
In Realm	<input type="text"/>
Out Realm	<input type="text"/>
In SA	<input type="text"/>
Out SA	<input type="text"/>
In Source Addr	<input type="text"/>
Out Dest Addr	<input type="text"/>
In Network Interface	<input type="text"/>
Out Network Interface	<input type="text"/>

16. In the “**In Realm**” field, enter the name of the realm for which the incoming call belongs. Valid values are alpha-numeric characters. For example, access.
17. In the “**Out Realm**” field, enter the name of the realm for which the outgoing call belongs. Valid values are alpha-numeric characters. For example, backbone.
18. In the “**In SA**” field, enter the name of the session agent (SA) on the incoming call session. Valid values are alpha-numeric characters. For example, SA1.
19. In the “**Out SA**” field, enter the name of the session agent (SA) on the outgoing call session. Valid values are alpha-numeric characters. For example, SA2.
20. In the “**In Source Addr**” field, enter the source IP address of the SA that accepted the incoming call session. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 172.45.6.7.
21. In the “**Out Dest Addr**” field, enter the destination IP address of the SA that accepted the outgoing call session. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 172.64.56.7.
22. In the In Network Interface field, enter the incoming core network interface that connects the Oracle Communications Session Border Controller to your network. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 192.45.6.7.

23. In the Out Network Interface field, enter the outgoing network interface that connects your Oracle Communications Session Border Controller to the outside network. IP Address must be entered in dotted decimal format (0.0.0.0). For example, 192.45.6.8.
24. Click **<Search>** to perform the search with the values you specified. A list of the records that the search process filtered, display in the window.

The Web-based GUI saves the search specifications until you click **<Reset>** in the search dialog box, OR until you log out of the Web-based GUI.

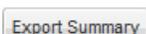
Exporting Information to a Text File

The Web-based GUI allows you to export information to a text file from the Sessions, Registrations, Subscriptions and Notable Events Reports, as well as from a specific ladder diagram, or from a page containing the results of a search. The data exports to a file that you can open and view as required.

You can export any of the following:

- All information from each report
- Information from a specific record only
- Information from a search result
- Information from a Ladder Diagram

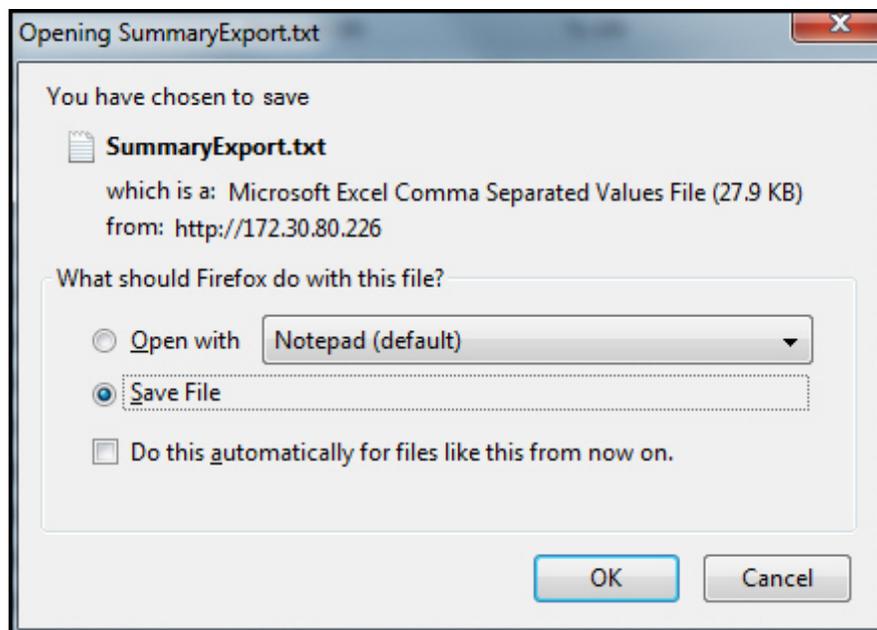
The following table identifies the buttons to use to export specific information from the Web-based GUI. All the export buttons in the GUI export to text files.

Button	Description
From the Sessions, Registrations, Subscriptions, and Notable Events Reports:	
	Exports the SIP messages and media events associated with the selected session, to a file in text format on the local machine.
	Exports all logged session summary records to a file in text format on the local machine. Note: This button exports ALL call session summary records or the records that matched a search criteria to the file.
From the Ladder Diagram:	
	Exports all of the information in the Ladder Diagram (Session Summary, SIP Message Details, and QoS statistics), to an HTML file format on the local machine.
	Exports detailed information about the SIP messages and media events in the Ladder Diagram associated with the selected session, to a file in text format on the local machine.

The following procedure is an example of using the export buttons in the Web-based GUI:

To export information to a text file:

1. In the applicable Report window or Ladder Diagram window, click the appropriate Export button. The following example illustrates that the **<Export Summary>** button was clicked on a report page. The following prompt displays.



The Web-based GUI assigns the file name for the text file (shown as SummaryExport.txt in the illustration above.)

Note:

The Web-based GUI exports Ladder Diagrams as HTML files.

2. Click **“Open with”** and select the application for which to open the resulting text file. Alternatively, click Save File to save the text file to your local PC.
3. Click **<OK>** to export the session or media information to the text file.

The following illustration shows a partial summary of the Sessions Report exported to a text file and opened using Microsoft Word. For additional examples of exported files, see Appendix B.

```

Export Name:
-----
Date: 2012-01-10 10:10:10
Time: 10:10:10
From: 10.10.10.10
To: 10.10.10.10
Duration: 00:00:00
Status: Success
-----

```

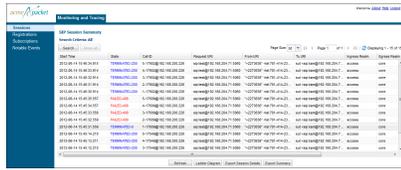
Session Reports

The Session Report is a SIP session summary of all logged call sessions on the Oracle Communications Session Border Controller. The data that displays in this table is dependent on the filters set via the ACLI on the Oracle Communications Session Border Controller.

Note:

For setting filters to capture applicable data, see Filters to Configure.

The columns that display on the Session Report page are dependent on the columns you selected in the procedure, [Customizing the Page Display](#).

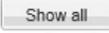


The following table describes the columns on this page.

Heading	Description
Start Time	Timestamp of the first SIP message in the call session.
State	Status of the call or media session. Valid values are: INITIAL Session for which an INVITE or SUBSCRIBE was forwarded. EARLY Session received the first provisional response (1xx other than 100). ESTABLISHED Session for which a success (2xx) response was received. TERMINATED Session that has ended by receiving or sending a BYE for an “Established” session or forwarding an error response for an “Initial” or Early session. The session remains in the terminated state until all the resources for the session are freed up. FAILED Session that has failed due to a 4xx or 5xx error code.
Call ID	Identification of the call source. Includes the phone number and source IP address.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource via a protocol, name, location, and any other applicable characteristic, and is sent by the Oracle Communications Session Border Controller in REQUEST headers.
To URI	URI formatted string that identifies the call destination information.
From URI	URI formatted string that identifies the call source information.
Ingress Realm	Incoming realm name.
Egress Realm	Outgoing realm name.
Duration	Amount of time, in seconds, that the call or media event was active.
Notable Event	Indicates if a notable event has occurred on the call session. Valid values are: short session - Sessions that don’t meet a minimum configurable duration threshold; Session dialogue, captured media information and termination signalling; Any event flagged as a short session interesting event local rejection - Sessions locally rejected at the Oracle Communications Session Border Controller for any reason (for example, Session Agent (SA) unavailable, no route found, SIP signalling error, etc.); Session dialogue, capture media information and termination signalling; Any event flagged as a local rejection interesting event
Session ID	Identification assigned to the call session.
Ingress Src Addr	Source IP address of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.
Calling Pkts	Number of packets that occurred during outbound calling media traffic.
Called Pkts	Number of packets that occurred during inbound called media traffic.
Calling R Factor	Average Quality of Service (QoS) factor observed during the incoming call session. Quality of service shapes traffic to provide different priority and level of performance to different data flows. R-factors are metrics in VoIP, that use a formula to take into account both user perceptions and the cumulative effect of equipment impairments to arrive at a numeric expression of voice quality. This column defines the call or transmission quality expressed as an R factor.

Heading	Description
Called RFactor	QoS factor observed during the outgoing call session. Quality of service shapes traffic to provide different priority and level of performance to different data flows. R-factors are metrics in VoIP, that use a formula to take into account both user perceptions and the cumulative effect of equipment impairments to arrive at a numeric expression of voice quality. This column defines the call or transmission quality expressed as an R factor.
Calling MOS	A measure of voice quality for an incoming media stream.
Called MOS	A measure of voice quality for an outgoing media stream.

The following table describes the buttons on this page.

Button	Description
	Performs a refresh of the page and updates it with the latest data.
	Displays a Ladder Diagram of a specific record in the table. The Ladder Diagram displays detailed information about a call session or media event.
	Exports the SIP messages and media events associated with the selected session, to a file in text format on the local machine.
	Exports all logged session summary records to a file in text format on the local machine.
	Allows you to specify parameters for performing a search for specific session summary records within the current report.
	Displays all of the session summary records in the Session Report.

For information about the Export buttons on this page, see [Exporting Information to a Text File](#).

Session Report Configuration

To display a Session Report:

1. After logging into the Web-based GUI, click Sessions in the left column. The session summary report displays.
2. Use the buttons on the bottom of the page to view information about the records in this report.

For information about Refresh, see [Refresh](#).

For information about the Ladder Diagram, see [Ladder Diagram](#).

For information about exporting information, see [Exporting Information to a Text File](#).

Ladder Diagram

Ladder diagrams in the Web-based GUI are logical schematics that show the call and media flow of packets on ingress and egress routes via the Oracle Communications Session Border Controller.

Ladder diagrams for the Session Report display the following session summary information:

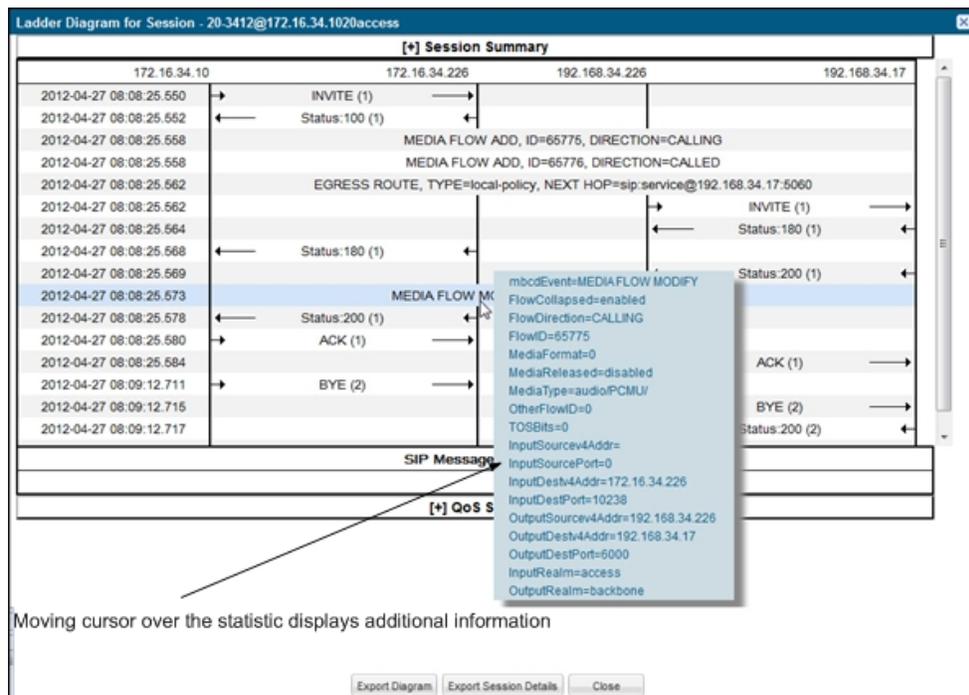
- Quality of Service (QoS) statistics for call sessions
- SIP messages and media events in time sequence

To display a ladder diagram for a specific record in the Session Report, you can double-click a record in the summary table OR click <Ladder Diagram> on the Session Report page.



To display a ladder diagram:

1. On the Sessions Report page, click <Ladder Diagram>. Alternatively, select a record in the table and double-click on that record. The following is an example of the ladder diagram that displays.



The Session Record Ladder Diagram consists of the following information:

- **Session Summary** - summary information about the call or media session in focus. For more information, see [Session Summary](#).
- **SIP Message Details** - SIP message and call flow information about the call or media session in focus. For more information, see [SIP Message Details](#).
- **QoS Statistics** - Quality of Service (QoS) statistic information about the call or media session in focus. For more information, see [QoS Statistics](#).

You can move your mouse over any statistic in the Ladder Diagram to view additional parameters and associated values for the statistic in a pop-up window.

The following table describes the buttons in this Ladder Diagram window.

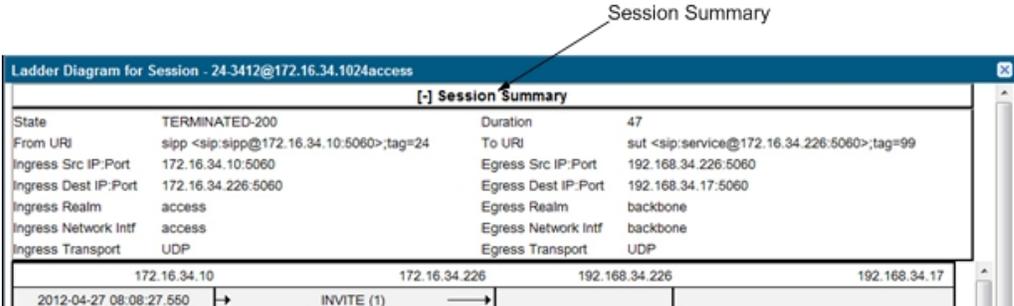
Button	Description
	Exports all of the information in the Ladder Diagram (Session Summary, SIP Message Details, and QoS statistics), to a file in text format on the local machine.
	Exports detailed information about the SIP messages and media events associated with the session in focus, to a file in text format on the local machine.
	Closes the Ladder Diagram window.

Note:

For information about the Export buttons on this page, see [Exporting Information to a Text File](#).

Session Summary

The Session Summary window in the Ladder Diagram displays an overall summary of the call or media session in focus.



Session Summary

[-] Session Summary			
State	TERMINATED-200	Duration	47
From URI	sipp <sip:sipp@172.16.34.10:5060>;tag=24	To URI	sut <sip:service@172.16.34.226:5060>;tag=99
Ingress Src IP:Port	172.16.34.10:5060	Egress Src IP:Port	192.168.34.226:5060
Ingress Dest IP:Port	172.16.34.226:5060	Egress Dest IP:Port	192.168.34.17:5060
Ingress Realm	access	Egress Realm	backbone
Ingress Network Intf	access	Egress Network Intf	backbone
Ingress Transport	UDP	Egress Transport	UDP

172.16.34.10 172.16.34.226 192.168.34.226 192.168.34.17

2012-04-27 08:08:27.550 → INVITE (1) →

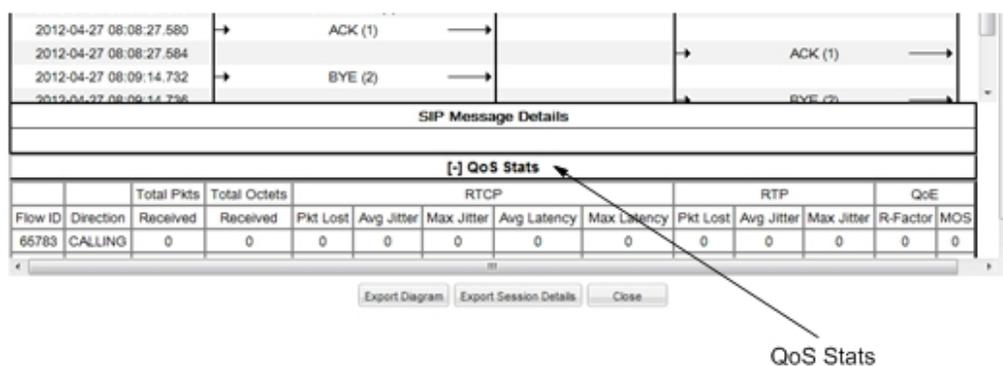
To display the Session Summary:

1. In the Ladder Diagram, click the [+] next to Session Summary at the top of the Ladder Diagram window. The Session Summary window expands. This window displays a summary of information about the call or media session in focus. The following table describes each field in the Session Summary window.

or media event. You can use the information to troubleshoot calls/media events that failed or timed out when trying to connect.

QoS Statistics

The Quality of Service (QoS) window displays information about the quality of the service used on the call session or media event when the call or event was active.



To display QoS Statistics:

1. In the Ladder Diagram, click the [+] next to QoS Stats at the bottom of the Ladder Diagram window. The QoS window expands. This window displays the QoS statistics for the call session or media event in focus. The following table describes each field in the QoS Statistics window.

Heading	Description
Flow ID	ID number assigned to the call session or media event flow of data.
Direction	The direction of the call or media event flow. Valid values are: CALLING (egress direction) CALLED (ingress direction)
Total Pkts Received	Total number of data packets received on the interface during the active call session or media event.
Total Octets Received	Total number of octets received on the interface during the active call session or media event. An octet is a unit of digital information that consists of eight bits.
RTCP	Real-time Transport Control Protocol - used to send control packets to participants in a call.
Pkts Lost	Number of RTCP data packets lost on the interface during the active call session or media event.
Avg Jitter	Average measure of the variability over time of the RTCP packet latency across a network. A network with constant latency has no variation (or jitter). Jitter is referred to as Packet Delay Variation (PDV). It is the difference in the one-way end-to-end delay values for packets of a flow. Jitter is measured in terms of a time deviation from the nominal packet interarrival times for successive packets.
Max Jitter	Maximum measure of the variability over time of the RTCP packet latency across a network. A network with constant latency has no variation (or jitter).

Heading	Description
Avg Latency	Average observed one-way signaling latency during the active window period. This is the average amount of time the signaling travels in one direction.
Max Latency	Maximum observed one-way signaling latency during the sliding window period. This is the maximum amount of time the signaling travels in one direction. Note: For more information about a sliding window period, see the <i>Oracle Communications Historical Data Recording Resource Guide</i> .
RTP	Real-Time Transport Protocol - a standard packet format for delivering audio and video over the internet.
Pkts Lost	Number of RTP data packets lost on the interface during the active call session or media event.
Avg Jitter	Average measure of the variability over time of the RTP packet latency across a network. A network with constant latency has no variation (or jitter). Jitter is referred to as Packet Delay Variation (PDV). It is the difference in the one-way end-to-end delay values for packets of a flow. Jitter is measured in terms of a time deviation from the nominal packet interarrival times for successive packets.
Max Jitter	Maximum measure of the variability over time of the RTP packet latency across a network. A network with constant latency has no variation (or jitter).
QoE	Quality of Experience - measurement used to determine how well the network is satisfying the end user's requirements.
R-Factor	Average Quality of Service (QoS) factor observed during the active window period. Quality of service shapes traffic to provide different priority and level of performance to different data flows. R-factors are metrics in VoIP, that use a formula to take into account both user perceptions and the cumulative effect of equipment impairments to arrive at a numeric expression of voice quality. This statistic defines the call or transmission quality expressed as an R factor.
MOS	Mean Opinion Score (MOS) score. MOS is a measure of voice quality. MOS gives a numerical indication of the perceived quality of the media received after being transmitted and eventually compressed using Codecs.

- Click [-] to close the QoS Stats window.

Registration Reports

The Registration Report is a summary of all logged SIP registrations sessions on the Oracle Communications Session Border Controller. The data that displays in this table is dependent on the filters set via the ACLI on the Oracle Communications Session Border Controller.



Note:

For setting filters to capture applicable data, see [Filters to Configure](#).

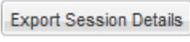
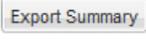
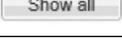
The columns that display on the Registration Report page are dependent on the columns you selected in the procedure, [Customizing the Page Display](#).



The following table describes the columns on this page.

Heading	Description
Start Time	Timestamp of the first SIP message in the call session.
Call ID	Identification of the call source. Includes the phone number and source IP address.
To URI	URI formatted string that identifies the call destination information.
From URI	URI formatted string that identifies the call source information.
Local Expires	The current setting for the expiration of a registration request sent from the Integrated Media Gateway (IMG) to a Remote SIP User Agent. The default is 3600 sec.
Remote Expires	The current setting for the expiration of a registration request sent from the Remote SIP User Agent to the Integrated Media Gateway (IMG). The default is 3600 sec.
Ingress Realm	Incoming realm name.
Egress Realm	Outgoing realm name.
Notable Event	Indicates if a notable event has occurred on the call session. Valid value is: local rejection - Sessions locally rejected at the Oracle Communications Session Border Controller for any reason (for example, Session Agent (SA) unavailable, no route found, SIP signalling error, etc.); Session dialogue, capture media information and termination signalling; Any event flagged as a local rejection interesting event
Session ID	Identification assigned to the call session.
Ingress Src Addr	Source IP address of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource via a protocol, name, location, and any other applicable characteristic, and is sent by the system in REQUEST headers.

The following table describes the buttons on this page.

Button	Description
	Displays a Ladder Diagram of a specific record in the table. The Ladder Diagram displays detailed information about a call registration.
	Exports the call registration information associated with the selected record, to a file in text format on the local machine.
	Exports all logged call registration records to a file in text format on the local machine.
	Performs a refresh of the page and updates it with the latest data.
	Allows you to specify parameters for performing a search for specific call registration records within the current report.
	Displays all of the call registration records in the Registrations Report.

To display a Registrations Report:

1. After logging into the Web-based GUI, click Registrations in the left column. The registrations summary report displays.

- Use the buttons on the bottom of the page to view information about the records in this report. For information about Refresh, see [Refresh](#). For information about the Ladder Diagram, see [Ladder Diagram](#). For information about exporting information, see [Exporting Information to a Text File](#).

Subscription Reports

The Subscription Report is a summary of all logged SIP subscription sessions on the Oracle Communications Session Border Controller. The data that displays in this table is dependent on the filters set via the ACLI on the Oracle Communications Session Border Controller.



Note:

For setting filters to capture applicable data, see [Filters to Configure](#).

The columns that display on the Subscription Report page are dependent on the columns you selected in the procedure, [Customizing the Page Display](#).

The following table describes the columns on this page.

Heading	Description
Start Time	Timestamp of the first SIP message in the call session.
Call ID	Identification of the call source. Includes the phone number and source IP address.
From URI	URI formatted string that identifies the call source information.
To URI	URI formatted string that identifies the call destination information.

Heading	Description
Events	<p>Specific subscribe event package that was sent from an endpoint to the destination endpoint. Applicable event packages can be:</p> <p>conference - Event package that allows users to subscribe to a conference Uniform Resource Identifier (URI).</p> <p>consent-pending additions - Event package used by SIP relays to inform user agents about the consent-related status of the entries to be added to a resource list.</p> <p>dialog - Event package that allows users to subscribe to another user, and receive notifications about the changes in the state of the INVITE-initiated dialogs in which the user is involved.</p> <p>kpml - Event package that enables monitoring of dual-tone multi-frequency (DTMF) signals, and uses XML documents called Key Press Markup Language (KPML).</p> <p>message-summary - Event package that carries message-waiting status and message summaries from a messaging system to an interested User Agent (UA).</p> <p>presence - Event package that conveys the ability and willingness of a user to communicate across a set of devices. A presence protocol is a protocol for providing a presence service over the Internet or any IP network.</p> <p>reg - Event package that provides a way to monitor the status of *all* the registrations for a particular Address of Record (AoR).</p> <p>refer - Event package that provides a mechanism to allow the party sending the REFER to be notified of the outcome of a referenced request.</p> <p>.winfo - Event package for watcher information. It tracks the state of subscriptions to a resource in another package.</p> <p>vq-rtcp - Event package that collects and reports the metrics that measure quality for RTP sessions.</p>
Local Expires	The current setting for the expiration of a registration request sent from the Integrated Media Gateway (IMG) to a Remote SIP User Agent. The default is 3600 sec.
Remote Expires	The current setting for the expiration of a registration request sent from the Remote SIP User Agent to the Integrated Media Gateway (IMG). The default is 3600 sec.
Ingress Realm	Incoming realm name.
Egress Realm	Outgoing realm name.
Notable Event	Indicates if a notable event has occurred on the call session. Valid value is: local rejection - Sessions locally rejected on the system for any reason (for example, Session Agent (SA) unavailable, no route found, SIP signalling error, etc.); Session dialogue, capture media information and termination signalling; Any event flagged as a local rejection interesting event
Session ID	Identification assigned to the call session.
Ingress Src Addr	Source IP address of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource via a protocol, name, location, and any other applicable characteristic, and is sent by the system in REQUEST headers.

The following table describes the buttons on this page.

Button	Description
	Displays a Ladder Diagram of a specific record in the table. The Ladder Diagram displays detailed information about a call subscription.

Heading	Description
State	<p>Status of the call or media event session. Valid values are:</p> <p>INITIAL Session for which an INVITE or SUBSCRIBE was forwarded.</p> <p>EARLY Session received the first provisional response (1xx other than 100).</p> <p>ESTABLISHED Session for which a success (2xx) response was received.</p> <p>TERMINATED Session that has ended by receiving or sending a BYE for an “Established” session or forwarding an error response for an “Initial” or Early session. The session remains in the terminated state until all the resources for the session are freed up.</p> <p>FAILED Session that has failed due to a 4xx or 5xx error code.</p>
Call ID	Identification of the call source. Includes the phone number and source IP address.
Request URI	Uniform Resource Identifier (URI) formatted string that identifies a resource via a protocol, name, location, and any other applicable characteristic, and is sent by the system in REQUEST headers.
To URI	URI formatted string that identifies the call destination information.
From URI	URI formatted string that identifies the call source information.
Events	<p>Specific subscribe event package that was sent from an endpoint to the destination endpoint. Applicable event packages can be:</p> <p>conference - Event package that allows users to subscribe to a conference Uniform Resource Identifier (URI).</p> <p>consent-pending additions - Event package used by SIP relays to inform user agents about the consent-related status of the entries to be added to a resource list.</p> <p>dialog - Event package that allows users to subscribe to another user, and receive notifications about the changes in the state of the INVITE-initiated dialogs in which the user is involved.</p> <p>kpml - Event package that enables monitoring of dual-tone multi-frequency (DTMF) signals, and uses XML documents called Key Press Markup Language (KPML).</p> <p>message-summary - Event package that carries message-waiting status and message summaries from a messaging system to an interested User Agent (UA).</p> <p>presence - Event package that conveys the ability and willingness of a user to communicate across a set of devices. A presence protocol is a protocol for providing a presence service over the Internet or any IP network.</p> <p>reg - Event package that provides a way to monitor the status of *all* the registrations for a particular Address of Record (AoR).</p> <p>refer - Event package that provides a mechanism to allow the party sending the REFER to be notified of the outcome of a referenced request.</p> <p>.winfo - Event package for watcher information. It tracks the state of subscriptions to a resource in another package.</p> <p>vq-rtcp - Event package that collects and reports the metrics that measure quality for RTP sessions.</p>
Ingress Realm	Incoming realm name.
Egress Realm	Outgoing realm name.
Notable Event	<p>Indicates if an interesting event has occurred on the call session. Valid values are:</p> <p>short session - A session is viewed as a "short session" if the length of time, in seconds, is equal to or below the short-session-duration value configured at the path Terminal->session-router->session-router-config->short-session-duration in the ACLI.</p> <p>local rejection - Sessions that are locally rejected at the Oracle Communications Session Border Controller for any reason (for example, Session Agent (SA) unavailable, no route found, SIP signalling error, etc.).</p>
Session ID	Identification assigned to the call session.

Heading	Description
Ingress Src Addr	Source IP address of the incoming call or media event.
Egress Dest Addr	Destination IP address of the outgoing call or media event.

The following table describes the buttons on this page.

Button	Description
	Displays a Ladder Diagram of a specific record in the table. The Ladder Diagram displays detailed information about a notable event.
	Exports the notable event information associated with the selected record, to a file in text format on the local machine.
	Exports all logged notable event records to a file in text format on the local machine.
	Performs a refresh of the page and updates it with the latest data.
	Allows you to specify parameters for performing a search for specific notable event records within the current report.
	Displays all of the notable event records in the Notable Events Report.

To display a Notable Events Report:

1. After logging into the Web-based GUI, click Notable Events in the left column. The notable events summary report displays.
2. Use the buttons on the bottom of the page to view information about the records in this report. For information about Refresh, see [Refresh](#). For information about the Ladder Diagram, see [Ladder Diagram](#). For information about exporting information, see [Exporting Information to a Text File](#).