Oracle® Communications Application Session Controller

Release Notes

Release 3.7.0

May 2016

Contents

NNOS-E
Release Notes, Version 3.7.0

This notice describes the current release of the Oracle OS-E® software. Systems running the OS-E software provide we-based application integration with IP communications as well as application level security, control, monitoring, and interoperability services for SIP-based Unified Communications.

Note: For existing customers who are upgrading from a prior release, the Covergence software components have been renamed under Oracle, Inc., as follows:

- Covergence—Now **Oracle**
- Session Manager—Now **Oracle Communications OS-E** (**OS-E**)
- CMS Web—Now **OS-E Management System**
- CXC-354—Now **OS-E**
- CXC-554—Now **OS-E**
- CVA—Now **Net-Net OS-E Virtual Machine** (**OS-E VM**)
- CLI prompt—Now **NNOS-E** (default)

You should review this notice for details about Release 3.7.0, information about operational considerations and known issues from prior releases, and for instructions on installing and upgrading to this release.

Technical Documentation

The Net-Net OS-E references in this documentation apply to the Net-Net OS-E operating system software that is used for the following Oracle and third-party SBC products:

- Oracle Communications Application Session Controller (ASC)
- Oracle Communications WebRTC Session Controller (WSC)
- Oracle Communications OS-E Session Director (SD) Session Border Controller (SBC)
- Oracle Communications 2600 Session Director (SD) Session Border Controller (SBC)
- Third-party products that license and use Oracle Communications OS-E software on an OEM basis

Unless otherwise stated, references to Net-Net OS-E in this document apply to all of the Oracle and third-party vendor products that use Net-Net OS-E software.

Oracle provides the following documentation set in PDF format, viewable using Adobe Reader 5.0 or later. These PDF files are available when you download OS-E software from Oracle, from your customer Web portal, as well as from docs.acmepacket.com.

- *Oracle Communications Application Session Controller System and Installation Commissioning Guide*
- *Oracle Communications Application Session Controller System and Installation Commissioning Guide Release 3.7.0M4*
- *Oracle Communications Application Session Controller Management Tools*
- *Oracle Communications Application Session Controller System Administration Guide*
- *Oracle Communications Application Session Controller Session Services Configuration Guide*
- *Oracle Communications Application Session Controller Objects and Properties Reference*
- *Oracle Communications Application Session Controller System Operations and Troubleshooting*

- *Oracle Communications Application Session Controller Release Notes*
- *Oracle Communications Application Session Controller Single Number Reach Application Guide*
- *Oracle Communications Application Session Controller Web Services SOAP REST API*
- *Oracle Communications WebRTC Session Controller Installation Guide*

Note: Oracle provides updates to the manuals on a regular basis. Go to your Oracle Web portal for the latest files in PDF format.

Release Note Revision History

This section contains a revision history for this document.

Date	Revision Number	Description
June 28, 2013	Revision 1.00	• Initial Release of 3.7.0 software.
December 30, 2013	Revision 1.01	• Provides a list of the OS-E system files available for download.

Date	Revision Number	Description
October 20, 2015	Revision 1.02	• Adds “Diffie-Hellman Logjam Attack Defense” to the “Known Problems, Restrictions, and Operational Considerations in 3.7.0” section.
May 17, 2016	Revision 1.03	• Removes an unnecessary step from the “Upgrading To Release 3.7.0 From Release 3.6.0M5” section. • Adds the *Oracle Communications Application Session Controller System Installation and Commissioning Guide Release 3.7.0M4* to the 3.7.0 doc set. • Adds “Installing and Upgrading Release 3.7.0M4” section. • Adds a note to the list of “OS-E System Files Available For Download” regarding the files available for release 3.7.0M4. • Removes the following sections which are all available in the Oracle Communications Application Session Controller System Installation and Commissioning Guide: “Assigning a Management IP Address”, “Using the Setup Script”, “Logging on Using the OS-E Management System”, “Building the Configuration File”, “Creating Cluster Networks”, “Installing the OS-E Software Updates”, “Information on OS-E Licensing”, and “Interoperating With SIP Vendors”. • Removes the “CDR Field Descriptions and Data Types” and “RADIUS Attributes, CDRs and RADIUS Servers” sections which are available in the *Oracle Communications Application Session Controller System Administration* guide. • Removes the “New Event Log Messages” and “Downloading Optional Management Files” sections which are available in the *Oracle Communications Application Session Controller Management Tools* guide.

Third-Party Platforms and Blades

The OS-E software is supported on third-party platforms and blades as part of integrated solutions that can only be obtained through authorized OEM partners.

For additional details, contact your Oracle representative.

OS-E Virtual Machine

The OS-E software, available as a VMware® or Xen virtual machine (VM), runs on x86-based PCs and servers.

For complete information on downloading and running the VM on compatible x86-based PCs and servers, refer to the Oracle Communications OS-E Installation and Commissioning Guide.

Software Download and Commissioning Process

The software download mechanism allows new and existing customers to acquire OS-E software directly from Oracle. Using secure URLs that can be accessed over the internet, Oracle provides all necessary software downloads for USB creation, product licensing, and commissioning of your selected hardware.

As part of each download, and depending on your actual requirements, Oracle provides the following:

- Oracle Boot Media Creator and the OS-E Release 3.7.0 software.
- Feature licenses.
- Documentation on how to create a OS-E USB stick for commissioning the software on your selected hardware.
- Standard set of OS-E technical publications.

If not included in the shipment, you will need to provide a USB stick with between 1-4GB storage to handle OS-E software downloads. Oracle has tested a variety of USB sticks available from current suppliers and manufacturers. Most USB sticks manufactured today will work.

For complete information on accessing the Oracle download server, creating an installation USB, and commissioning a OS-E device, refer to the Oracle Communications OS-E Installation and Commissioning Guide.

OS-E System Files Available for Download

The OS-E system files available for individual download are as follows:

- Oracle Communications Application Session Controller E3.7.0[m*x*] Installation Image Supertar
- Oracle Communications Application Session Controller E3.7.0[m*x*] Installation USB image
- Oracle Communications Application Session Controller E3.7.0[m*x*] Installation ISO image
- Oracle Communications Application Session Controller E3.7.0[m*x*] VMWare VMX/VMDK file
- Oracle Communications Application Session Controller E3.7.0[m*x*] Xen server image
- Oracle Communications Application Session Controller E3.7.0[m*x*] HyperV OVA file
- Oracle Communications Application Session Controller E3.7.0[m*x*] LCR Import Tool
- Oracle Communications Application Session Controller E3.7.0[m*x*] Embedded LCR Import Tool
- Oracle Communications Application Session Controller E3.7.0[m*x*] Samples Kit
- Oracle Communications Application Session Controller E3.7.0[m*x*] Archive Viewer Application
- Oracle Communications Application Session Controller E3.7.0[m*x*] Weblogic SDK file
- Oracle Communications Application Session Controller E3.7.0[m*x*] License Document

Note: This list applies to all 3.7.0[m*x*] releases except for 3.7.0M4. For an accurate list of 3.7.0M4 OS-E system files available for individual download, see the *Oracle Communication Application Session Controller 3.7.0 Maintenance Release Guide*.

Installing and Upgrading Release 3.7.0M4

As of release 3.7.0M4, the OS-E runs on Oracle Linux (version 7.0 and higher) as opposed to its own custom kernel as in prior releases. Because the OS-E runs on Oracle Linux and uses a yum to install and update RPM files, the installation and upgrade process has changed significantly.

Due to this fact, a 3.7.0M4 version of the installation guide has been created. The *Oracle Communications Application Session Controller System Installation and Commissioning Guide Release 3.7.0M4* is now available with the 3.7.0 doc set.

Upgrading to Release 3.7.0

This section explains now to upgrade to Release 3.7.0 from previous releases of the OS-E.

Upgrading To Release 3.7.0 From Release 3.6.0m5

From Release 3.6.0m5, perform the upgrade to Release 3.7.0 using the following procedure.

Note: Oracle recommends you run this procedure from a local console so you do not lose connectivity during the procedure. If you choose to run the upgrade remotely over SSH or from the OS-E Management user interface **Tools** option, you are unable to monitor the progress of the upgrade process.

Note: In order to provide a full set of ASC Samples in the 3.7.0 installation, the size of the overall installation package has increased beyond 1Gb.

Because of this, you now need a 4Gb USB installer disk image to 4Gb (for example, a blue Kanguru Oracle stick).

To upgrade to 3.7.0 from 3.6.0m5:

1. From the current configuration, save the standard configuration using a unique file name of your choice with the .cfg extension. Preserve this file as a backup copy of your configuration in case you need to revert to the earlier release.

   For example:

```
NNOS-E>> config save standard cxcbackup.cfg
Success
```

2. Save the configuration as an XML file.

   For example:

```
NNOS-E>> config save xml /cxc_common/cfg370.xml
Success
```

3. Copy or run SCP to copy the following file to the /cxc directory on the OS-E.

   - nnSE370.tar.gz

   Note: When logged in using the OS-E Management user interface, you can use the **Tools/ Upload File** option to browse for the file on your local PC or network location to obtain the file to upload to the OS-E.

4. At the NNOS-E prompt, run the **install** action, as follows:

```
NNOS-E> install file nnSE370.tar.gz
Are you sure (y or n)? y
Installing: nnSE370.tar.gz
Success! Rebooting Session Manager
```

   Note: When logged in using the CMS Web to perform the upgrade remotely, use the CMS **Tools/ Update Software** option with **Install the Update** checked off to run the upgrade.

Upgrading To Release 3.7.0 From Release 3.6.0, 3.6.0m1, 3.6.0m2, 3.6.0m3, and 3.6.0m4

From a release earlier than 3.6.0m5, perform the upgrade to Release 3.7.0 using the following procedure.

Note: Oracle recommends you run this procedure from a local console so you do not lose connectivity during the procedure. If you choose to run the upgrade remotely over SSH or from the OS-E Management user interface **Tools** option, you are unable to monitor the progress of the upgrade process.

To upgrade to 3.7.0:

1. From the current configuration, save the standard configuration using a unique file name of your choice with the .cfg extension. Preserve this file as a backup copy of your configuration if you need to revert to the earlier release.

   For example:

```
NNOS-E>> config save standard cxcbackup.cfg
Success
```

2. Save the configuration as an XML file.

   For example:

```
NNOS-E>> config save xml /cxc_common/cfg360.xml
Success
```

3. Copy or run SCP to copy the following file to the /cxc directory on the OS-E.

   - nnSE370.tar.gz

   If you are currently logged on using the OS-E Management user interface, you can use the **Tools/ Upload File** option to browse for the file on your local PC or network location to obtain the file to upload to the OS-E.

4. At the NNOS-E prompt, run the **install** action, as follows:

```
NNOS-E> install file nnSE370.tar.gz
Are you sure (y or n)? y
Installing: nnSE370.tar.gz
Success! Rebooting Session Manager
```

5. Convert the Release 3.6 configuration using the following style sheets:

   - 3.6.0-to-3.6.0.m3.xsl
   - 3.6.0.m3-to-3.6.0.m4.xsl

```
NNOS-E>> xml transform 3.6.0-to-3.6.0.m3.xsl /cxc_common/cxc360.xml /
   cxc_common/cxc360.m3.xml
Success!
NNOS-E>> xml transform 3.6.0.m3-to-3.6.0.m4.xsl /cxc_common/
   cxc360m3.xml /cxc_common/cxc360m4.xml
Success!
```

6. Replace the configuration file with the new file named cxc360m4.xml.

```
NNOS-E>> config replace cxc360m4.xml
```

7. Save the configuration in standard format using the default configuration file name (cxc.cfg).

```
NNOS-E>> config save standard cxc.cfg
```

8. Perform a restart warm to boot with the new configuration.

```
NNOS-E>> restart warm
```

If you are using the CMS Web to perform the upgrade remotely, use the CMS **Tools/ Update Software** option with the “Install the Update” selection checked off to run the upgrade.

Upgrading To Release 3.7.0 From Release 3.5.x

The accounting functionality changed from release 3.5.x to release 3.6.0. When purging is enabled, the OS-E accounting service deletes all 3.5.x unpersisted CDRs it finds during the upgrade process.

In order to not lose any raw CDRs, you must find accounting targets which have not received all applicable CDR data. Using the **scan utility** action you can view an accounting store to see if there are any unpersisted CDRs in the 3.5.x version that need to be persisted before the upgrade to 3.6.0 can be done. The execution of the **scan utility** action is performed in the shell.

1. Copy or run SCP to copy the following file to the /cxc directory on OS-E.

- accounting35store

If you are currently logged on using the OS-E Management user interface, you can use the **Tools/Upload File** option to browse for the file on your local PC or network location to get the file uploaded to OS-E.

2. Stop running traffic. This allows any lagging accounting service to catch up.

3. At the NNOS-E> prompt, enter the **shell** action as follows:

```
NNOS-E> shell
localhost app_slot_1 #
```

4. Enter the scan utility command using the following syntax:

```
localhost app_slot_1 # accounting35store -s [3.5x store location] -m
   [target]
```

where:

- -s—Specifies the accounting store folder. The location of this folder is specified in the configuration under the **accounting-root-directory**.
- -m—Specifies the mode for results which can be **target** (results per target), **file** (results per file), or **incomplete** (list incomplete files only; default behavior)

For example:

```
localhost app_slot_1 # accounting35store -s /acme_common/accounting35/
   test1/ -m target
```

The following are more arguments you can use with the **scan utility** to narrow down your accounting target search:

- -f—Specifies an individual file to check
- -csv—Output the results in CSV format
- -w—The results are printed into this file.
- -force—Scan every file
- -d—Specifies the debug mode which can be **error** (only errors are printed; default), **info** (information level messages are printed), or **debug** (debug level messages are printed)
- -h—Displays the help for the scan utility.

5. If any bad accounting targets are found, fix them. The following is an example of the message you receive when a bad target is found:

```
localhost app_slot_1 # accounting35store -s /acme_common/accounting35/
   test1/ -m target
-----------------------------------------------------------------
***  Failed Targets ***
-----------------------------------------------------------------
  1. vsp\accounting\database\group mssqlDB
        server-name:
               type: database
             failed: 24
-----------------------------------------------------------------
```

6. Allow accounting service to run until processing is complete. To ensure the process is complete you can execute the **scan utility** again.

   After all of this has been completed, you can begin the upgrade process.

If you are currently running Release 3.5.x, perform the upgrade to Release 3.6.0 using the procedure covered in the Net-Net OS-E 3.6.0 Release Notes, available from the Oracle Web portal.

Note: Oracle recommends you run this procedure from a local console so you do not lose connectivity during the procedure. If you choose to run the upgrade remotely over SSH or from the OS-E Management user interface **Tools** option, you are unable to monitor the progress of the upgrade process.

To upgrade to 3.7.0:

1. From the current configuration, save the standard configuration using a unique file name of your choice with the .cfg extension. Preserve this file as a backup copy of your configuration if you need to revert to the earlier release.

   For example:

```
NNOS-E>> config save standard cxcbackup.cfg
Success
```

2. Save the configuration as an XML file.

   For example:

```
NNOS-E>> config save xml /cxc_common/cxc35.xml
Success
```

3. Copy or run SCP to copy the following files to the /cxc directory on OS-E.

   - nnSE370.tar.gz

   If you are currently logged on using the OS-E Management user interface, you can use the **Tools/Upload File** option to browse for the file on your local PC or network location to get the file uploaded to OS-E.

4. At the NNOS-E> prompt, run the **install** program, as follows:

```
NNOS-E>> install file nnSE370.tar.gz
Are you sure (y or n)? y
Installing: nnSE370.tar.gz
Success! Rebooting Net-Net OS-E
```

5. 5. Convert the Release 3.5 configuration using the following style sheets:

   - • 3.5-to-3.6.xsl
   - • 3.6.0-to-3.6.0m3.xsl
   - • 3.6.0m3-to-3.6.0m4.xsl

```
NNOS-E>> xml transform 3.5-to-3.6.xsl /cxc_common/cxc35.xml /
   cxc_common/cxc360.xml
Success

NNOS-E>> xml transform 3.6.0-to-3.6.0.m3.xsl /cxc_common/cxc360.xml /
   cxc_common/cxc360m3.xml
Success!

NNOS-E>> xml transform 3.6.0.m3-to-3.6.0.m4.xsl /cxc_common/
   cxc360m3.xml /cxc_common/cxc360m4.xml
Success!
Installing: nnSE360m4.tar.gz
Success! Rebooting Net-Net OS-E
```

6. Replace the configuration file with the new file named cxc36m4.xml

```
NNOS-E>> config replace cxc36m4.xml
```

7. Save the configuration in standard format using the default configuration file name (cxc.cfg).

```
NNOS-E>> config save standard cxc.cfg
```

8. Perform a restart warm to boot with the new configuration.

```
NNOS-E>> restart warm
```

If you are using the OS-E Management System to perform the upgrade remotely, use the **Tools/Update Software** option with the “Install the Update” selection checked off to run the upgrade.

Upgrading To Release 3.7.0 From Release 3.4.x or later

If you are currently running Release 3.4.2, 3.4.3 or 3.4.4, you should perform the upgrade to Release 3.7.0 from a USB stick. Refer to the Oracle Communications OS-E Installation and Commissioning Guide for information on creating the USB stick and commissioning the OS-E device. Contact Oracle for assistance when performing these upgrades.

Note: When upgrading from release 3.4 to releases 3.5, 3.6, or 3.7 on third party hardware, a manual procedure must be performed to ensure that the interfaces remain on the current ethernet ports.

Special Considerations After Running the Upgrade

OS-E creates an alternate, inactive directory that captures the files associated with the release from which you are upgrading. This inactive directory holds customer-created configuration files and phone configurations. You may need to access this directory to copy these custom files to the new active release directory. Otherwise, for example, SIP phones may not work properly.

The release files associated with the older release are moved to a directory of the form:

/cxc_rel/app-<slot 1-3>

where "app" is a literal text string, followed by the version and release numbers that are explicit to the release software.

Oracle recommends that you place a copy of any uploaded configuration or phone files into a common directory for easy access when upgrades are completed. For example, copy the files into the /cxc_common/ directory so that the files remain there after any upgrade.

Release 3.7.0

This section describes all of the new adaptations added to the OS-E in release 3.7.0, including new features, configuration objects and properties, and MIBs.

New Features

- WebRTC Support
- Configuring Authorization
- Configuring Route Headers In an SMX Environment
- Recreating the Route Server Default Route File Name
- Enabling HTTP Authentication for the Route Server Import Tool
- Route Server Variable Support
- Upgraded 64-Bit Kernel
- Supporting Multiple Host Names on a Virtual Host HTTPS Certificate
- Custom Scheme Support in SIP Headers
- SIP Proxy Enhancements
- Whitelist and Blacklist Permissions Filtering
- Mitigating Application-Level DoS Attacks on OS-E HTTP Services
- Call-Control Action Updates
- Boot Media Creator Updates

WebRTC Support

What is WebRTC?

WebRTC is an open source technology standard that enables browser to browser communications for voice, video, and P2P file sharing without the need for plugins.

WebRTC implements three JavaScript APIs:

- getUserMedia—Get access to data streams
- RTCPeerConnection—Audio or video calling with facilities for encryption and bandwidth management

- RTCDataChannel—Peer-to-peer communication of generic data

While WebRTC does not include any standards for signaling, the OS-E supports two types of WebRTC signaling:

- WebRTC using SIP signaling over websockets
- WebRTC using OS-E Call Control REST APIs

In addition to supporting WebRTC, the OS-E can also act as a multimedia streaming server (MSS). The MSS allows SIP and H.323 endpoints to communicate over web-based multimedia applications using either a third-party Flash media server or directly on the OS-E as an internal media server. For more information on how to configure the MSS, see the Configuring the Multimedia Streaming Server chapter in this guide.

WebRTC Media Handling

Like SIP endpoints, WebRTC endpoints use Session Description Protocol (SDP) as a means to exchange media capabilities. Using the WebRTC APIs, a browser can access user's cameras and microphones and transmit these media streams over the network. In order to provide secure and reliable transmission across a variety of network topologies, all WebRTC endpoints must support both Interactive Connectivity Establishment (ICE) and Secure Real-Time Transport Protocol (SRTP) in their SDP exchanges. For more information on SDP, visit http://tools.ietf.org/html/rfc4566.

What is ICE?

ICE is a protocol that establishes network paths for UDP-based media streams. It is an extension of the SDP offer/answer model and works by discovering and including all possible media transport addresses (known as candidates) in the SDP. Once SDPs are exchanged, ICE tests all possible media paths using the Session Traversal Utilities for the NAT (STUN) protocol as connectivity checks. Once the connectivity checking completes, the ICE agents settle on a final candidate pair to use for media transmission. The OS-E supports ICE on a per call-leg basis, meaning it can act as both the offering and answering ICE agent to satisfy this WebRTC requirement. For more information on ICE, visit http://tools.ietf.org/html/rfc5245.

What is STUN?

In addition to connectivity checking, ICE relies heavily on STUN to discover all possible media candidates. During this candidate gathering phase, ICE agents perform STUN requests to discover their public IP addresses when behind a NAT device. The OS-E can be configured as a STUN server to satisfy these initial STUN requests. For more information on STUN, visit http://www.ietf.org/rfc/rfc3489.

What is SDES-SRTP?

SRTP is secure RTP designed to provide encryption, authentication, and integrity to the RTP streams. In SDES-SRTP, encryption keys are exchanged in the SDP offer and answer using the crypto attribute. The OS-E supports SDES-SRTP encryption and decryption on a per call-leg basis to satisfy this WebRTC requirement. For more information on SDES, visit http://tools.ietf.org/html/rfc4568.

Configuring ICE and STUN

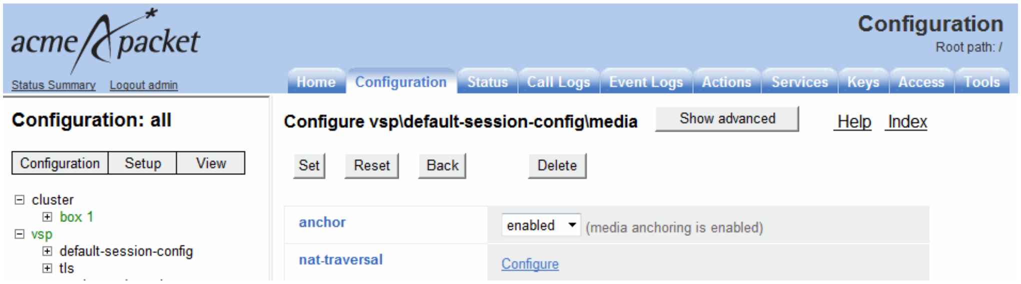
To configure ICE on the OS-E, you must enable session-wide media anchoring.

You must also enable symmetric RTP, which returns RTP based on the source IP address and UDP port in the received RTP. NAT modifies data in the IP header only and the SDP payload is left unchanged. By using the source IP address and UDP port from the received RTP, the OS-E sends traffic back to the NAT device instead of the untranslated addresses in the SDP.

In addition to these session-wide settings, you must also configure ICE for incoming and outgoing WebRTC sessions.

To enable system-wide media anchoring and symmetric RTP:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool** > **entry**.
2. Click **Configure** next to **media**.

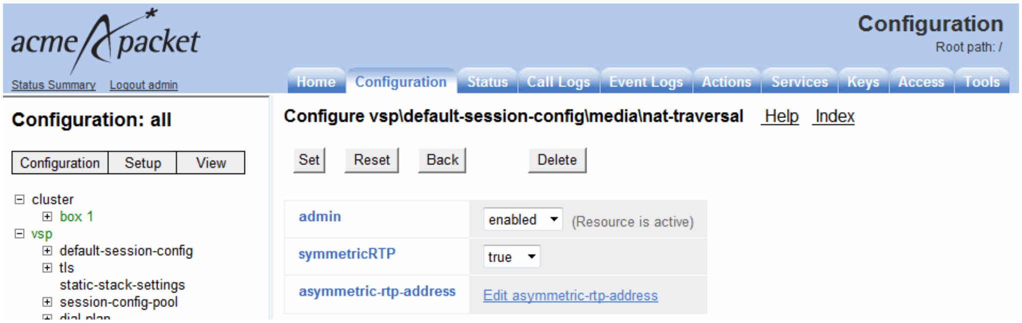


Configure vsp\default-session-config\media

anchor: enabled (media anchoring is enabled)

nat-traversal: Configure

3. **anchor**—Set to **enabled to** enable media anchoring for this media session. Media anchoring forces the SIP media session to traverse the OS-E.
4. Click **Configure** next to **nat-traversal**.



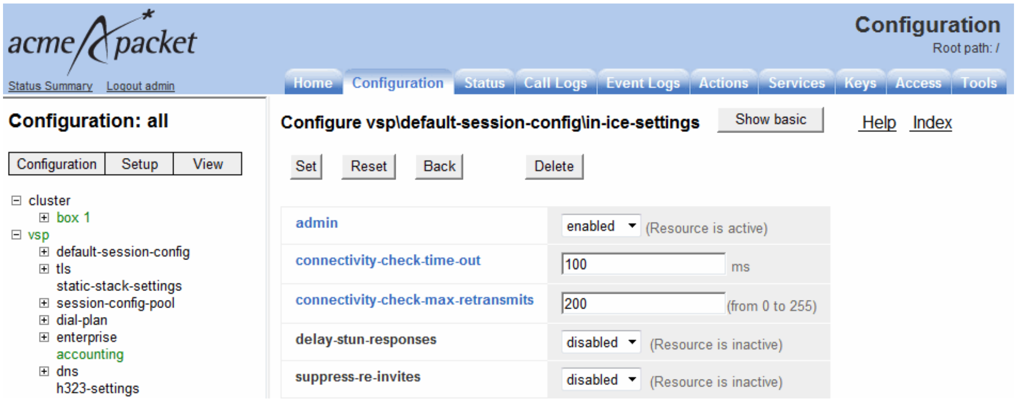
Configure vsp\default-session-config\media\nat-traversal

admin	enabled (Resource is active)
symmetricRTP	true
asymmetric-rtp-address	Edit asymmetric-rtp-address

5. **symmetricRTP**—Set to **true** to enable symmetric RTP for this media session. When enabled, symmetric RTP returns RTP based on the source IP address and UDP port in the received RTP. NAT modifies data in the IP header only and the SDP payload is left unchanged.
6. Click **Set**. You are returned to the **media** object.
7. Click **Set**. Update and save the configuration.

To enable ICE for incoming WebRTC sessions:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool** > **entry**.
2. Click **Configure** next to **in-ice-settings**.



Configure vsp\default-session-config\in-ice-settings

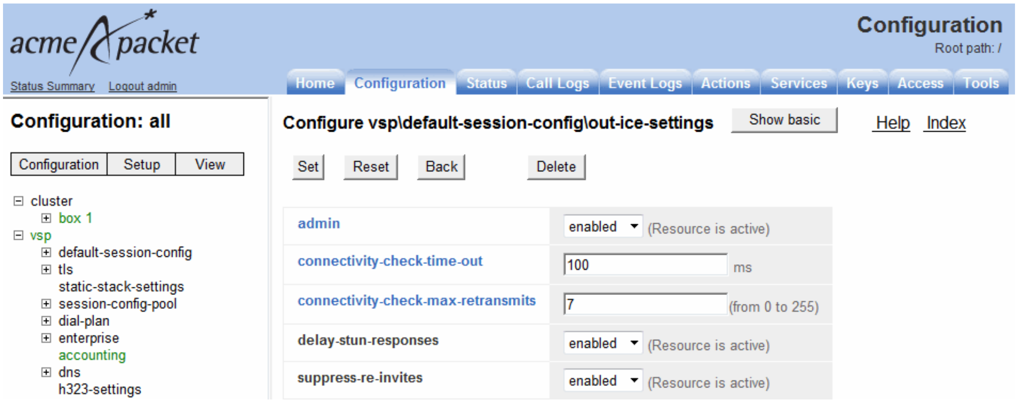
admin	enabled (Resource is active)
connectivity-check-time-out	100 ms
connectivity-check-max-retransmits	200 (from 0 to 255)
delay-stun-responses	disabled (Resource is inactive)
suppress-re-invites	disabled (Resource is inactive)

3. **admin**—Set to **enabled** to enable ICE on this call leg.

4. **connectivity-check-max-retransmits**—Specify the number of times the OS-E retransmits ICE STUN connectivity checks before labeling a candidate pair as Failed. To achieve maximum interoperability with Chrome, set this value to no less than **200**.
5. Click **Set**. Update and save the configuration.

To enable ICE for outgoing WebRTC sessions:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool > entry**.
2. Click **Configure** next to **out-ice-settings**.



Configure vsp\default-session-config\out-ice-settings

admin	enabled	(Resource is active)
connectivity-check-time-out	100	ms
connectivity-check-max-retransmits	7	(from 0 to 255)
delay-stun-responses	enabled	(Resource is active)
suppress-re-invites	enabled	(Resource is active)

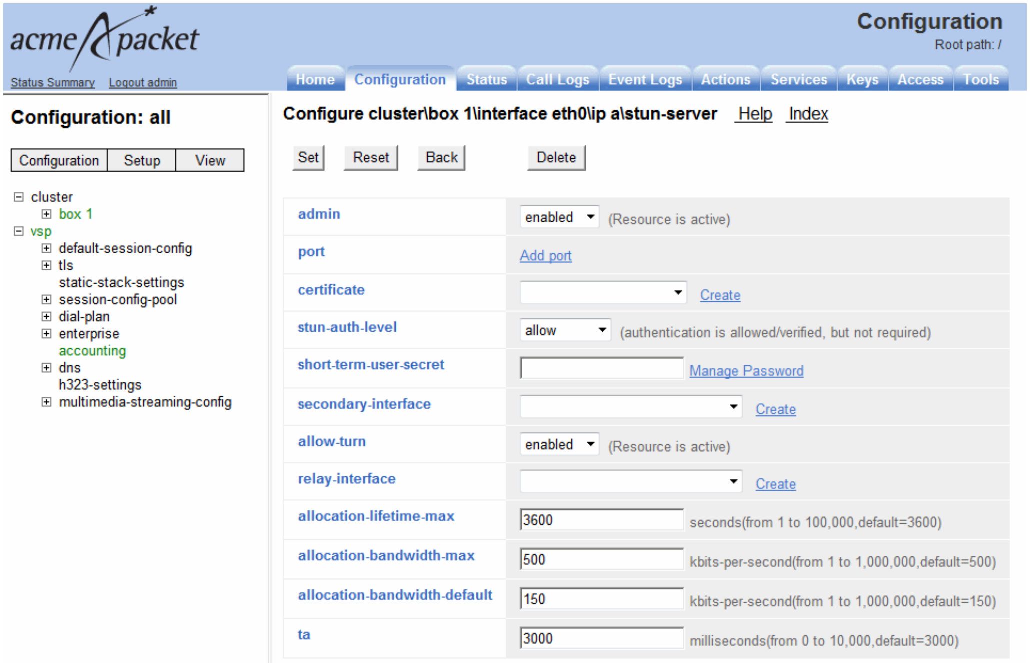
3. **admin**—Set to **enabled** to enable ICE on this call leg.
4. **delay-stun-responses**—*Advanced property*. Set to **enabled**. When enabled, the OS-E does not respond to STUN until the 200 OK is received.
5. **Note:** To view Advanced properties, you must click the **Show advanced** button.
6. **suppress-re-invites**—*Advanced property*. Set to **enabled**. When enabled, the OS-E does not send a re-INVITE when ICE completes successfully.
7. **Note:** To view Advanced properties, you must click the **Show advanced** button.
8. Click **Set**. Update and save the configuration.

Configuring STUN

In addition to an ICE server, the OS-E can also be configured as a STUN server.

To configure the OS-E as a STUN server:

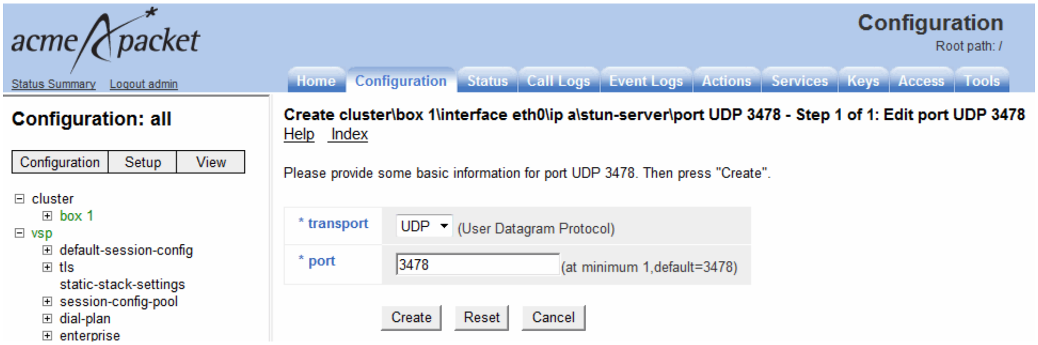
1. Click the **Configuration** tab and select the **cluster** > **box** > **interface** > **ip** object.
2. Click **Configure** next to **stun-server**.



Configure cluster\box 1\interface eth0\ip a\stun-server Help Index

admin	enabled (Resource is active)
port	Add port
certificate	Create
stun-auth-level	allow (authentication is allowed/verified, but not required)
short-term-user-secret	Manage Password
secondary-interface	Create
allow-turn	enabled (Resource is active)
relay-interface	Create
allocation-lifetime-max	3600 seconds(from 1 to 100,000,default=3600)
allocation-bandwidth-max	500 kbits-per-second(from 1 to 1,000,000,default=500)
allocation-bandwidth-default	150 kbits-per-second(from 1 to 1,000,000,default=150)
ta	3000 milliseconds(from 0 to 10,000,default=3000)

3. **admin**—Set to **enabled** to enable the OS-E as a STUN server.
4. Click **Add port** to configure a port for the STUN server.



Create cluster\box 1\interface eth0\ip a\stun-server\port UDP 3478 - Step 1 of 1: Edit port UDP 3478
Help Index

Please provide some basic information for port UDP 3478. Then press "Create".

* transport	UDP (User Datagram Protocol)
* port	3478 (at minimum 1,default=3478)

5. **transport**—Select from the drop-down list the transport protocol over which STUN messages are exchanged between a SIP endpoint and the OS-E STUN server. Valid values are **UDP**, **TCP**, and **TLS**. The default value is UDP.

6. **port**—Specify the port over which STUN messages are exchanged between a SIP endpoint and the OS-E STUN server. The default value is **3478**.
7. Click **Create**. You are returned to the **stun-server** object.
8. Click **Set**. Update and save the configuration.

For more information on the **stun-server** object, see the Oracle Communications OS-E Objects and Properties Reference Guide.

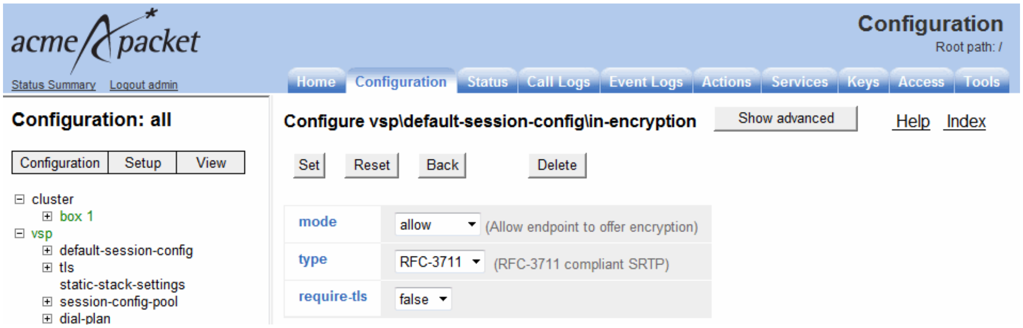
Configuring Encryption

Although the OS-E supports SDES encryption, it does not require it from WebRTC endpoints. If an endpoint does not support encryption, it does not include a crypto key in its answer SDP and RTP is automatically used to transport media.

Because the OS-E always sends media encrypted out, you must configure the in-leg to allow encryption and the out-leg to require it.

To configure in-leg encryption:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool** > **entry**.
2. Click **Configure** next to **in-encryption**.



Configure vsp\default-session-config\in-encryption

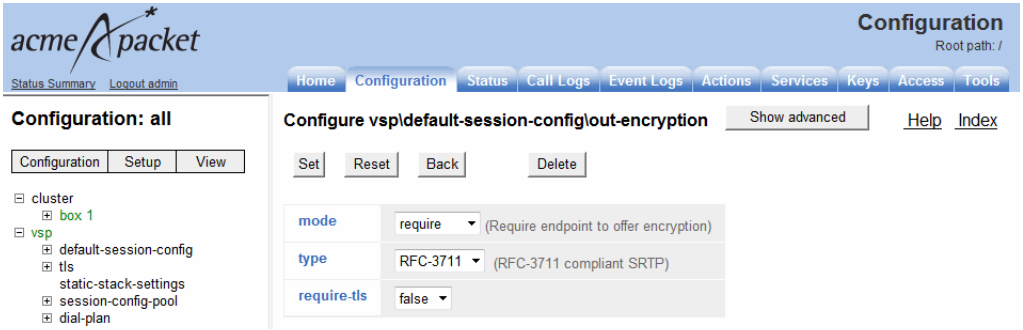
mode: allow (Allow endpoint to offer encryption)
type: RFC-3711 (RFC-3711 compliant SRTP)
require-tls: false

3. **mode**—Select **allow** from the drop-down list. This allows the OS-E to receive encryption on the in-leg.
4. Click **Set**. Update and save the configuration.

To configure out-leg encryption:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool** > **entry**.

2. Click **Configure** next to **out-encryption**.



Configure vsp\default-session-config\out-encryption

mode	require	(Require endpoint to offer encryption)
type	RFC-3711	(RFC-3711 compliant SRTP)
require-tls	false	

3. **mode**—Select **require** from the drop-down list. This allows the OS-E to offer encryption.

4. Click **Set**. Update and save the configuration.

RTP/RTCP Multiplexing

The OS-E supports RTP/RTCP multiplexing. When enabled, the OS-E bundles all of the RTP and RTCP media through the same port.

When initiating a bundled call, the OS-E inserts the necessary information into the INVITE message’s SDP in the following format:

```
m=RTP Port
a=rtcp=RTCP Port
a=rtcp-mux
```

If the recipient supports RTP/RTCP multiplexing, it returns the following in the SDP of its 200 OK response:

```
m=RTP/RTCP Port
a=rtcp-mux
```

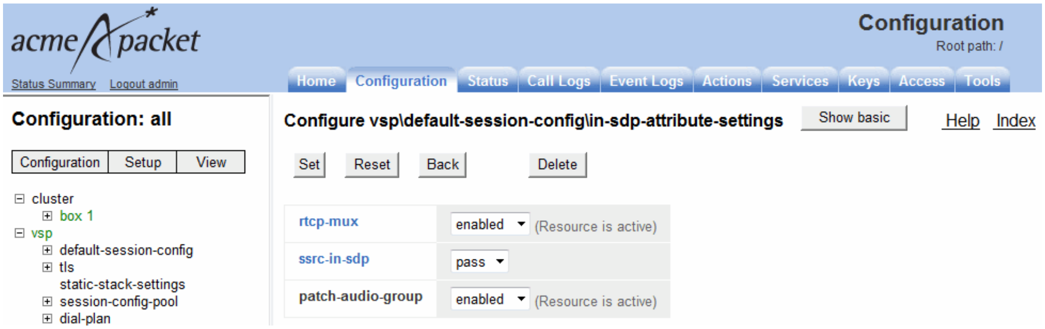
If the recipient does not support RTP/RTCP multiplexing, it returns its own RTP and RTCP port numbers in the SDP without a=rtcp-mux and multiplexing is not used.

The OS-E does not support audio and video multiplexing, which is audio and video streams bundled on the same port. To ensure the recipient the OS-E is talking to knows this, you must strip out any Synchronization Source (SSRC) information from the SDP.

To configure RTP/RTCP multiplexing for incoming WebRTC calls:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool** > **entry**.

2. Click **Configure** next to **in-sdp-attribute-settings**.



Configure vsp\default-session-config\in-sdp-attribute-settings

rtcp-mux	enabled (Resource is active)
ssrc-in-sdp	pass
patch-audio-group	enabled (Resource is active)

3. **rtcp-mux**—Enables or disables RTP/RTCP multiplexing. By default this is **disabled**.

4. **ssrc-in-sdp**—Set to **strip** to strip out any SSRC information from the SDP.

5. **patch-audio-group**—*Advanced property*. Set to **enabled**. When the OS-E receives an offer SDP with both audio and video and the line a=group BUNDLE audio video and a response with only audio, it must perform certain functions in order to get the audio to work.

   When enabled, the OS-E performs the following modifications:

   - The OS-E performs RTP/RTCP multiplexing on the in-leg, regardless of the user configuration

   - The OS-E adds bundling information by adding the following to the SDP

```
a=group BUNDLE audio
a=mid:audio
```

   - The OS-E generates WebRTC-style SSRC values and adds them to the SDP as well as the RTP/RTCP stream.

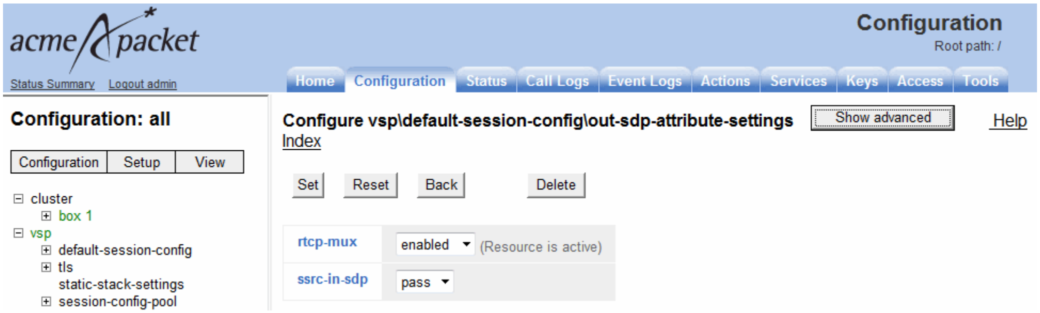
   Note: To view Advanced properties, you must click the **Show advanced** button.

6. Click **Set**. Update and save the configuration.

To configure RTP/RTCP multiplexing for outgoing WebRTC calls:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool** > **entry**.

2. Click **Configure** next to **out-sdp-attribute-settings**.



Configure vsp\default-session-config\out-sdp-attribute-settings

rtcp-mux	enabled (Resource is active)
ssrc-in-sdp	pass

3. **rtcp-mux**—Enables or disables RTP/RTCP multiplexing. By default this is **disabled**.
4. **ssrc-in-sdp**—Set to **strip** to strip out any SSRC information from the SDP.

   Note: To view Advanced properties, you must click the **Show advanced** button.
5. Click **Set**. Update and save the configuration.

Configuring SDP Regeneration

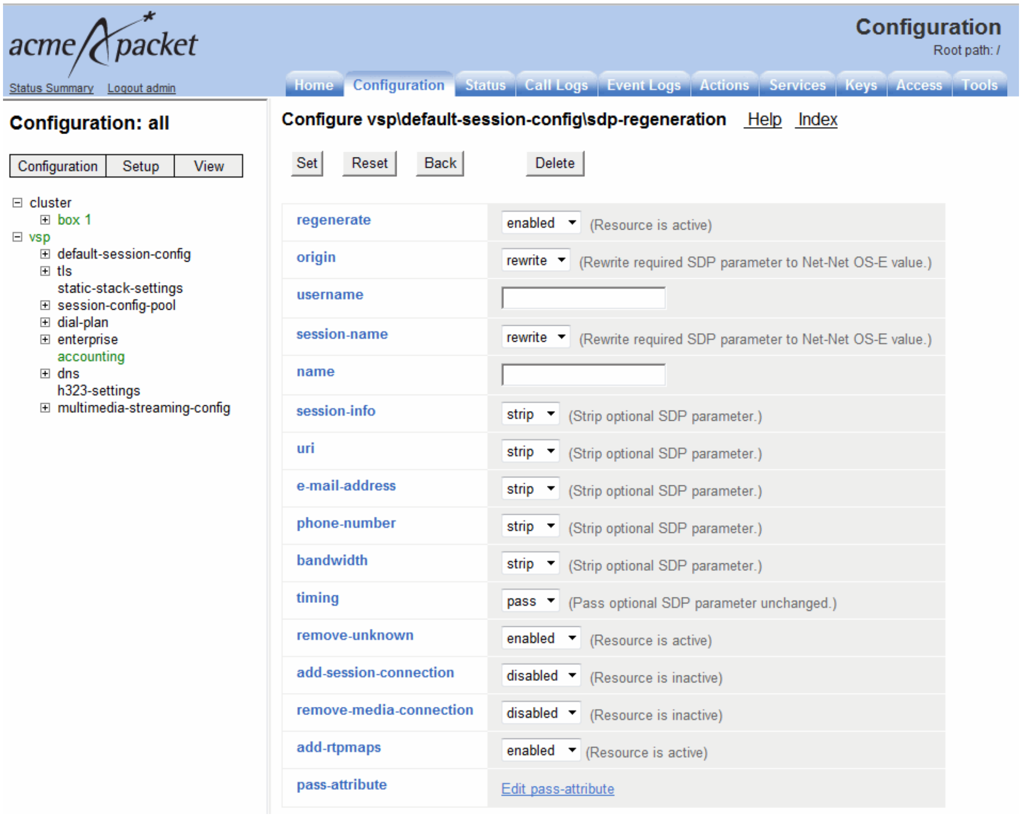
To ensure the OS-E represents itself properly in the SDP, it must regenerate incoming SDPs to list the attributes it supports and strip out unsupported attributes. To do this, you must configure the **sdp-regeneration** object.

> Note: If the OS-E forwards an SDP containing attributes it does not support, the WebRTC call will not work.

To configure SDP regeneration:

1. Click the **Configuration** tab and select either **default-session-config** or **session-config-pool > entry**.

2. Click **Configure** next to **sdp-regeneration**. The **sdp-regeneration** object appears.



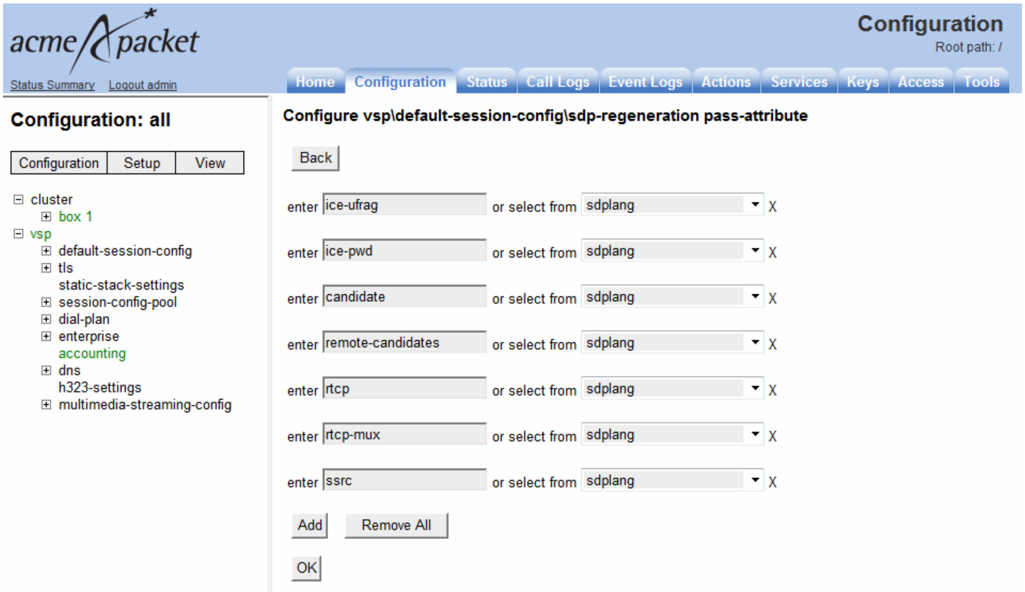
Configure vsp\default-session-config\sdp-regeneration

regenerate	enabled	(Resource is active)
origin	rewrite	(Rewrite required SDP parameter to Net-Net OS-E value.)
username		
session-name	rewrite	(Rewrite required SDP parameter to Net-Net OS-E value.)
name		
session-info	strip	(Strip optional SDP parameter.)
uri	strip	(Strip optional SDP parameter.)
e-mail-address	strip	(Strip optional SDP parameter.)
phone-number	strip	(Strip optional SDP parameter.)
bandwidth	strip	(Strip optional SDP parameter.)
timing	pass	(Pass optional SDP parameter unchanged.)
remove-unknown	enabled	(Resource is active)
add-session-connection	disabled	(Resource is inactive)
remove-media-connection	disabled	(Resource is inactive)
add-rtpmaps	enabled	(Resource is active)
pass-attribute	Edit pass-attribute	

3. **regenerate**—Set to **enabled** to regenerate the SDP, with the configured settings, before forwarding it along.

4. **add-rtpmaps**—Set to **enabled** so the OS-E includes rtpmap attributes for well-known CODECs when the rtpmap is not included in the SDP by the original endpoint.

5. **pass-attribute**—Click **Edit pass-attribute**.



Configure vsp\default-session-config\sdp-regeneration pass-attribute

enter ice-ufrag or select from sdplang X

enter ice-pwd or select from sdplang X

enter candidate or select from sdplang X

enter remote-candidates or select from sdplang X

enter rtcp or select from sdplang X

enter rtcp-mux or select from sdplang X

enter ssrc or select from sdplang X

6. Enter the attributes to be included in the SDP. The following attributes must be added:

   - ice-ufrag
   - ice-pwd
   - candidate
   - remote-candidates
   - rtcp
   - rtcp-mux
   - ssrc

You must enter attributes one at a time. After entering an attribute and clicking **Add**, a new field to enter the next attribute appears.

Note: These attributes do not appear in the drop-down list and must be entered into the provided blank field.

7. Click **OK**. You are returned to the **sdp-regeneration** object.

8. Click **Set**. Update and save the configuration.

Configuring WebRTC Using SIP Signaling Over WebSockets

One of the ways the OS-E implements signaling for WebRTC is via SIP over WebSockets. WebRTC applications can use JavaScript SIP stack APIs to perform signaling over a websocket and manages STUN requests and WebRTC media.

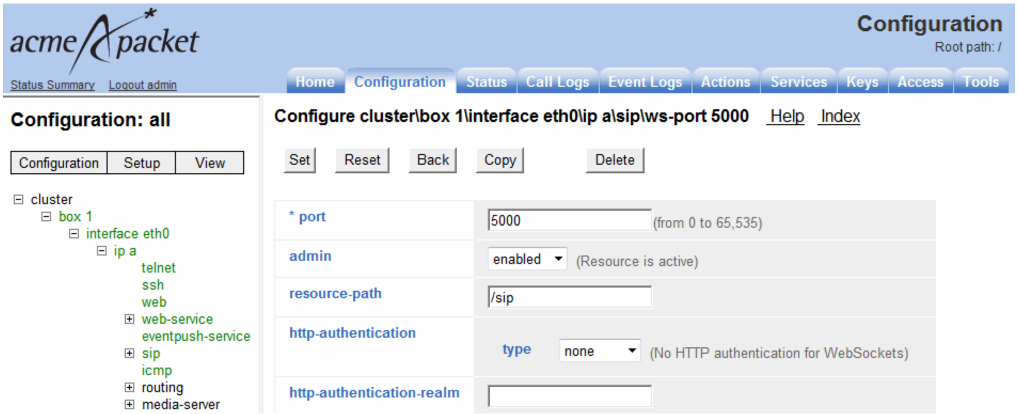
In addition to UDP, TCP, and TLS transport protocols, the OS-E supports two WebSocket-specific transport protocols. The ws-port is an unencrypted protocol and the wss-port is encrypted with TLS. When using wss-port, a certificate (configured under the **vsp** > **tls** > **certificate** object) is required.

SIP transport protocols can be configured as two types of sockets on the OS-E, listener sockets and outgoing connection sockets.

Configuring WebSocket Listener Sockets

To configure WebSocket listener sockets:

1. Click the **Configuration** tab and select the **cluster** > **box** > **interface** > **ip** object.
2. Click **Configure** next to **sip**.
3. Click **Add ws-port**.
4. **port**—Enter the port for the listener socket. There is no default port.
5. Click **Create**. The **ws-port** object appears.



Configure cluster\box 1\interface eth0\ip a\sip\ws-port 5000

* port	5000 (from 0 to 65,535)
admin	enabled (Resource is active)
resource-path	/sip
http-authentication	type none (No HTTP authentication for WebSockets)
http-authentication-realm	

6. **admin**—Specify whether this port is enabled or disabled. The default value is **enabled**.

7. **resource-path**—Specify the HTTP resource to expect in the HTTP GET message. The default value is **/sip**.
8. **http-authentication**—Specify the type of HTTP authentication, if any, that should be applied to the incoming GET message.
9. **Note:** No browser currently supports HTTP authentication of a WebSocket, so this property should be left as **none**.
10. **http-authentication-realm**—Specify the realm to use for HTTP authentication when enabled.
11. **Note:** Since no browsers currently support HTTP authentication of a WebSocket, this property should be left blank.
12. Click **Set**. Update and save the configuration.

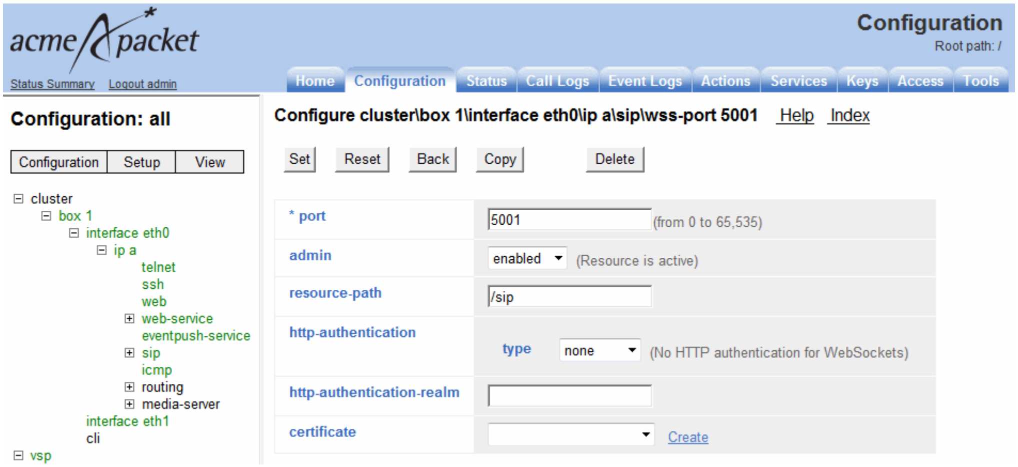
Configuring Secure WebSocket Listener Sockets

When you configure secure WebSocket listener sockets, you must upload a certificate to the **vsp > tls > certificate** object. For more information on the certificate object, see the Oracle Communications OS-E Objects and Properties Reference Guide.

To configure secure websocket listener sockets:

1. Click the **Configuration** tab and select the **cluster > box > interface > ip** object.
2. Click **Configure** next to **sip**.
3. Click **Add wss-port**.
4. **port**—Enter the port for the listener socket. There is no default port.

5. Click **Create**. The **wss-port** object appears.



Configure cluster\box 1\interface eth0\ip a\sip\wss-port 5001

* port	5001 (from 0 to 65,535)
admin	enabled (Resource is active)
resource-path	/sip
http-authentication	type none (No HTTP authentication for WebSockets)
http-authentication-realm	
certificate	Create

6. **admin**—Specify whether this port is enabled or disabled. The default value is **enabled**.

7. **resource-path**—Specify the HTTP resource to expect in the HTTP GET message. The default value is **/sip**.

8. **http-authentication**—Specify the type of HTTP authentication, if any, that should be applied to the incoming GET message.

9. **Note:** No browser currently supports HTTP authentication of a WebSocket, so this property should be left as **none**.

10. **certificate**—Select a certificate for this port from the drop-down list.

11. Note: This certificate is configured under the **vsp > tls > certificate** object. To configure a new certificate click **Create**. If no certificate is specified, the certificate the OS-E uses for SIP is used.

12. Click **Set**. Update and save the configuration.

WebRTC Using OS-E REST Call Control APIs

In WebRTC implementations using OS-E Call Control REST APIs, the web application instructs the OS-E to perform SIP signaling by calling OS-E call control APIs.

This section describes the OS-E APIs and their arguments used in the WebRTC implementation.

Note: Arguments surrounded by angle brackets (< >) are required and arguments surrounded by brackets ([ ]) are optional.

register

Executes a WebRTC call using the OS-E's REST APIs. This action allows you to bind a web endpoint to a particular URI. It creates a location cache entry and a unique binding that ties the remote application to the specified URI, allowing remote applications to start receiving calls for the URI without the need to statically configure a dial-plan that routes the calls to a web endpoint.

The URI is a SIP URI in the following formats:

```
sip:user@domain:port
```

When the **register** action is executed, the OS-E first verifies that the user has permission to register that URI. If not, the OS-E returns an “unauthorized” error message.

If the URI is valid, the OS-E performs a registration-plan lookup. If no matches are found, the OS-E returns a “no routes” error message. If a match is found, the OS-E creates a binding that ties the specified URI with the server returned by the registration-plan lookup. Along with the binding, the OS-E also creates an identifier that uniquely identifies the binding. This identifier persists throughout the lifetime of the binding. At the completion of a successful binding, the OS-E returns this identifier along with a “success” message.

When the URI sent in the register action is linked to an existing SIP server, the registration is executed asynchronously. The OS-E returns a “pending” message indicating that the application must monitor for a register event containing the result of the asynchronous registration.

Syntax

```
register <URI> [expiration]
```

Arguments

- <*URI*>—The URI tied to this binding.
- [*expiration*]—The expiration time of the binding in seconds. If not specified,

unregister

Disconnects a WebRTC call using OS-E’s REST APIs.

Syntax

```
unregister <URI> <binding-identifier>
```

Arguments

- <*URI*>—The URI tied to this binding.
- <*binding-identifier*>—The identifier that uniquely identifies this binding.

call-control-call

Initiates a call using provided To and From SIP URIs.

You can configure the OS-E to add post-dial digits to a call-control-call action by appending the string **postd=*digits*** to the user portion of the **To** argument.

Syntax

```
call-control-call <to> <from> [requestId] [originatorFirst]
[async] [transport] [config] [session-id] [content-type] [body]
```

Arguments

- <*to*>—The destination SIP URI of the session.
- <*from*>—The originating SIP URI of the session.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*originatorFirst*]—When **enabled** (the default), the originating party is connected first. When **disabled**, the called party is connected first.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*transport*]—The transport method to use for the call. This can be set to **any**, **TCP**, **UDP**, or **TLS**.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

Enclose the value in quotation marks when using the CLI.

- [*session-id*]—The optional session ID for the session.
- [*content-type*]—The content type of the message body of the initial call.
- [*body*]—The message body of the initial call.

call-control-ringing

Rings a destination to indicate an incoming call.

Syntax

```
call-control-ringing <handle> [content-type] [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-redirect

Redirects an initiated call to a new endpoint, prior to the call being answered. This creates a new call leg and cancels the original one.

Syntax

```
call-control-redirect <handle> <endpoint> [config]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*endpoint*>—The URI of the call’s destination.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

Enclose the value in quotation marks when using the CLI.

call-control-accept

Accepts an incoming call from an offering endpoint.

> **Note:** You must specify content-type as application/sdp and body as the SDP for the call.

Syntax

```
call-control-accept <handle> [content-type] [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-reject

Rejects an incoming call from an offering endpoint.

Syntax

```
call-control-reject <handle> [response-code] [responseText]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*response-code*]—The response code to be used in the response.
- [*responseText*]—The text to be included in the response.

call-control-transfer

Transfers the specified call leg to the specified To SIP URI. The original call leg, referred to by its handle, is disconnected. Handle can be thought of as belonging to the party doing the transfer, even though the transfer is done via a third-party action.

Syntax

```
call-control-transfer <handle> <to>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*to*>—The destination SIP URI of the session.

call-control-insert-dtmf

Inserts DTMF digits into SIP sessions. DTMF is inserted only into the call leg specified; the other party does not hear it.

Syntax

```
call-control-insert-dtmf <handle> <digits> [volume] [duration]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*digits*>—Specifies the digits inserted into the call leg.
- [*volume*]—The volume of the DTMF digits, in decimals from -36 to 0. The value **1** is the default.
- [*duration*]—The duration of each digit in milliseconds, from 100 to 10000. The value **0** is the default.

Creates a call to an endpoint from a given SIP URI. If you specify a From URI, it is used as the From URI in the SIP message; if you specify no From URI, the From URI is that of the given endpoint.

Syntax

```
call-control-park <endpoint> [from] [requestId] [async] [sessionID]
   [persist] [config]
```

Arguments

- <*endpoint*>—The URI of the call's destination.

- [*from*]—The originating SIP URI of the call.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled** (the default), the OS-E waits for the action to complete before returning a response.
- [*sessionID*]—The optional session ID for a rendezvous session.
- [*persist*]—When **enabled**, a connected session remains parked even when the remote endpoint disconnects.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

Enclose the value in quotation marks when using the CLI.

call-control-attach

Attaches a call leg to a an existing SIP session for three-way conferencing.

Syntax

```
call-control-attach <handle> <session-id>
```

Arguments

- <*handle*>—The handle of the endpoint to be attached.
- <*session-id*>—The session to which the endpoint is being attached.

call-control-disconnect

Terminates all parties in a SIP session.

Syntax

```
call-control-disconnect <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

Configuring Authorization

The OS-E’s security has been enhanced to allow you to configure authorization. Authorization consists of creating specific grants, or privileges.

There are three types of grants you can create:

- default-grants—Applies to all configured OS-E users
- attribute-grants—Applies to configured OS-E users based on values extracted from their attributes.
- group-grants—Applies to configured OS-E users based on group membership

The grants you can create apply to just a small segment of actions, which are divided into groups called resource-types. A resource-type is the OS-E function on which you are setting permissions.

The following table lists the resource types along with their corresponding actions.

Resource-Type	Associated Actions	CRUD Privileges
call	call-control-accept	
	call-control-annotate	
	call-control-attach	CU
	call-control-call	C
	call-control-call-to-session	CU
	call-control-connect	
	call-control-create-session	C
	call-control-destroy-session	D
	call-control-detach	D
	call-control-disconnect	D
	call-control-fork	U
	call-control-get-annotation	U
	call-control-hold	U

Resource-Type	Associated Actions	CRUD Privileges
	call-control-info-request	U
	call-control-intercept	U
	call-control-join	U
	call-control-message-request	U
	call-control-modify	U
	call-control-mute-off	U
	call-control-mute-on	U
	call-control-notify	U
	call-control-notify-request	U
	call-control-options-request	U
	call-control-park	CU
	call-control-park-to-session	CU
	call-control-persistence	U
	call-control-record-stop	C
	call-control-redirect	U
	call-control-reject	U
	call-control-retrieve	U
	call-control-ringing	U
	call-control-send-message	U
	call-control-subscribe-request	U
	call-control-terminate	D
	call-control-transfer	U
call-recording	call-control-record-start	C
	call-control-record-stop	C
call-monitor	call-control-monitor-file	CU
	call-control-monitor-session	CU
call-media-insertion	call-control-drop-file	CU
	call-control-insert-dtmf	U
	call-control-media-pause	CU
	call-control-media-resume	CU
	call-control-media-scanner-start	CU

Resource-Type	Associated Actions	CRUD Privileges
	call-control-media-scanner-stop	CU
	call-control-media-seek	CU
	call-control-media-stop	CU
	call-control-memo-begin	CU
	call-control-memo-end	CU
	call-control-play	U
sip-request	sip-send-message	CU
	sip-send-notify	CU
	sip-send-options	CU
	sip-send-other	CU
	sip-send-subscribe	CU
	sip-send-unsubscribe	CU
registration	register	C
	unregister	D
event-channel	dynamic-event-service register	CR
	dynamic-event-service keepalive	U
	dynamic-event-service unregister	D

In cases where an action has required either <*handle*> or <*session ID*> arguments, the OS-E extracts the To and From URI identities from each call leg, matches them against the resource-identity specified in a user’s privileges, and determines whether that user is authorized to perform an operation.

When configuring a grant, you must define privileges for that resource-type. Privileges specify what a user can or cannot do with that resource-type.

Privileges on the OS-E follow the standard CRUD model:

- create
- retrieve
- update
- delete

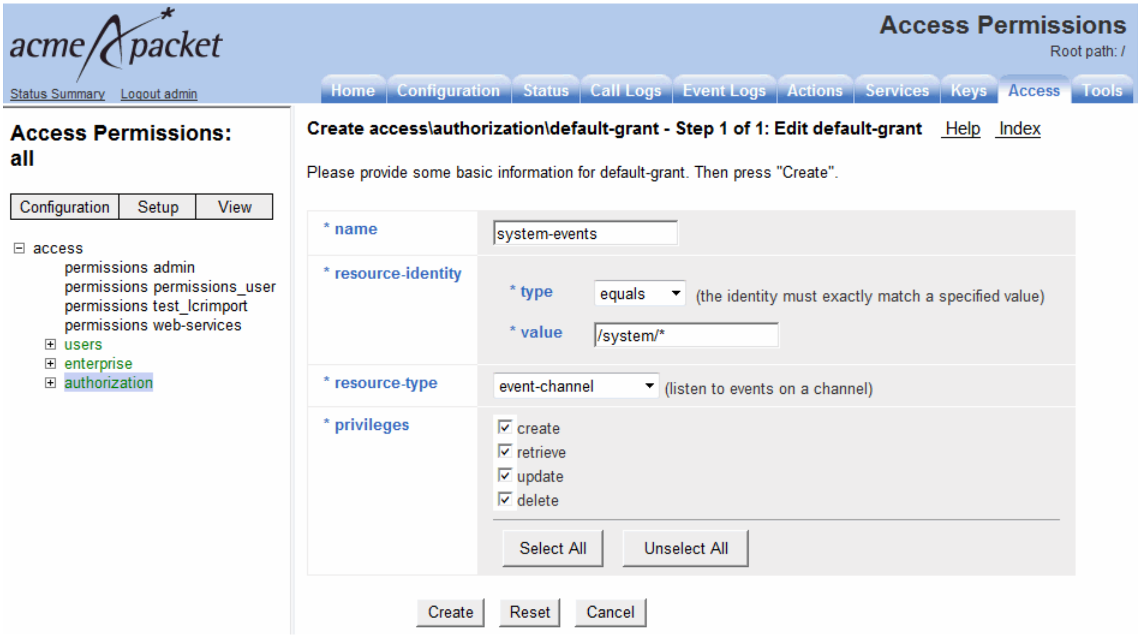
Configuring Default Grants

Configure grants under the Access tab’s **authorization** object.

Default grants are one of three types of grants you can configure on the OS-E. Default grants are grants that apply to all OS-E users matching the specified resource identity.

To configure default grants:

1. Select the **Access** tab and click **authorization**.
2. Set **admin** to **enabled** to enable authorization.
3. Click **Add default-grant**. The **default-grant** object appears.



Create access\authorization\default-grant - Step 1 of 1: Edit default-grant

Please provide some basic information for default-grant. Then press "Create".

* name	system-events
* resource-identity	* type: equals (the identity must exactly match a specified value) * value: /system/*
* resource-type	event-channel (listen to events on a channel)
* privileges	☑ create ☑ retrieve ☑ update ☑ delete

4. **name**—Enter a name to give this grant.
5. **resource-identity**—Select the type of matching to use to identify a resource-type. The following are valid values:
   - equals <*value*>—The value that a user provides during an authorization request must be exactly the same as the resulting resource-identity. This is the default setting.

- matches <*expression*>—The value that a user provides during an authorization request is matched against the resource-identity using a regular expression match.

  Note: For more information on using Regular Expressions, see the Oracle Communications OS-E Objects and Properties Reference Guide.

- any—Any value a user provides during an authorization request matches.

6. **resource-type**—Select the resource-type for this grant from the drop-down list.
7. **privileges**—Check the CRUD privileges to allow for this resource-type. By default, they are all selected.
8. Click **Create**.
9. Click **Set**. Update and save the configuration.

Configuring Attribute Grants

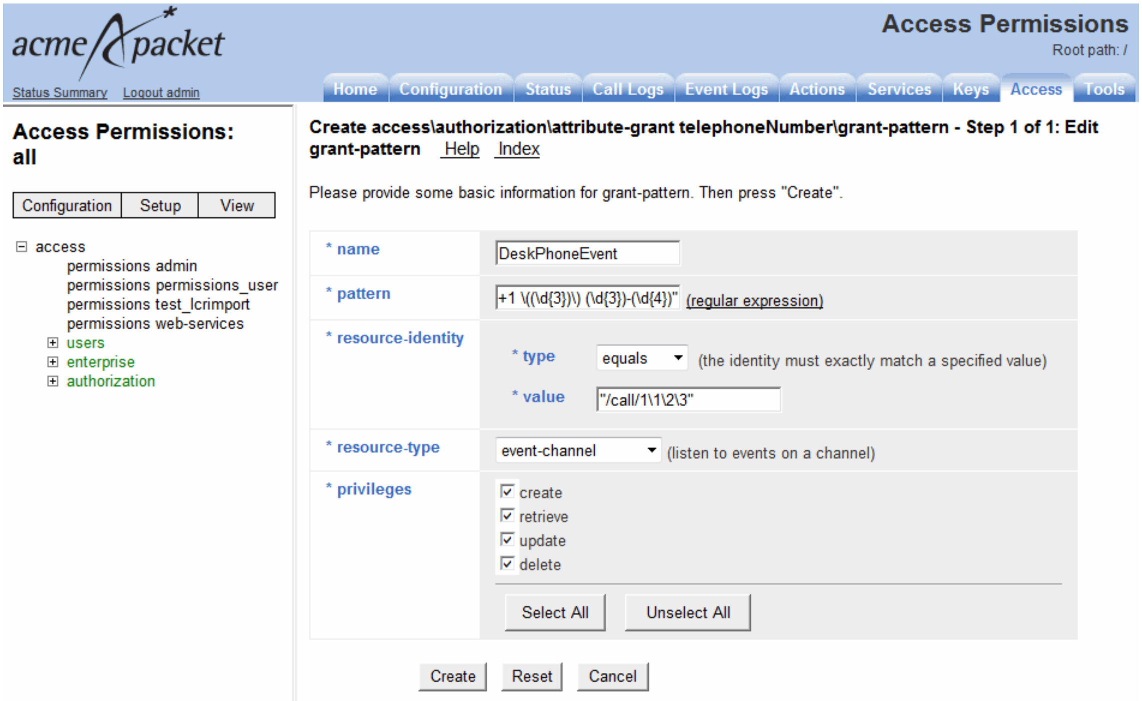
Attribute grants are grants that apply to all OS-E users that have the attribute and match the specified resource-identity.

To configure attribute-grants:

1. Select the **Access** tab and click **authorization**.
2. **name**—Enter the name of the attribute for which you are creating this grant.

   Note: The name you provide must be the name of an actual attribute used within the directory.
3. Click **Create**. The **attribute-grant** object appears.

4. Click **Add grant-pattern**.



Access Permissions: all

Create access\authorization\attribute-grant telephoneNumber\grant-pattern - Step 1 of 1: Edit grant-pattern

Please provide some basic information for grant-pattern. Then press "Create".

* name	DeskPhoneEvent
* pattern	+1 \((\d{3})\) (\d{3})-(\d{4})"
* resource-identity	* type: equals (the identity must exactly match a specified value) * value: "/call/1\1\2\3"
* resource-type	event-channel (listen to events on a channel)
* privileges	☑ create ☑ retrieve ☑ update ☑ delete

5. **name**—Enter a descriptive name to give this grant.

6. **pattern**—Enter the regular expression pattern to use to define the attribute.

7. **resource-identity**—Select the type of matching to use to identify a resource-type. The following are valid values:

   - equals <*value*>—The value that a user provides during an authorization request must be exactly the same as the resulting resource-identity. This is the default setting.

   - matches <*expression*>—The value that a user provides during an authorization request is matched against the resource-identity using a regular expression match.

   Note: For more information on using Regular Expressions, see the Oracle Communications OS-E Objects and Properties Reference Guide.

   - any—Any value a user provides during an authorization request matches.

8. **resource-type**—Select the resource-type for this grant from the drop-down list.

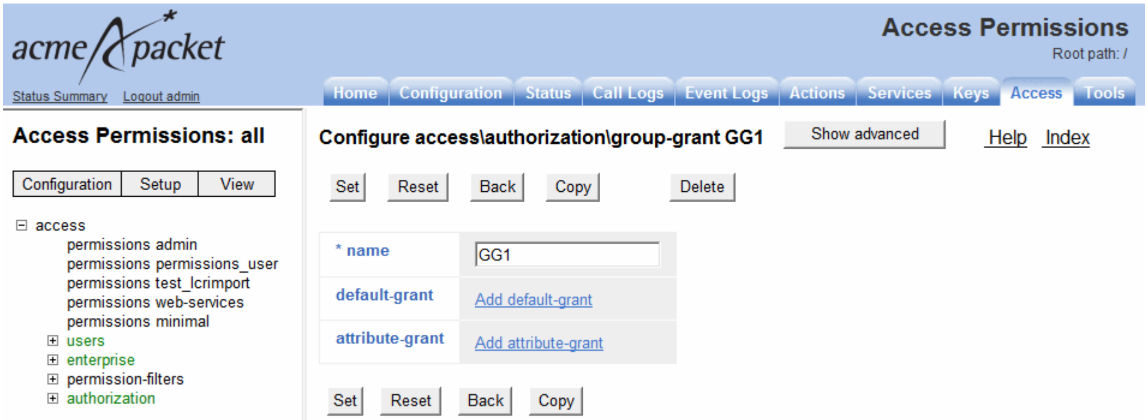
9. **privileges**—Check the CRUD privileges to allow for this resource-type. By default, they are all selected.
10. Click **Create**.
11. Click **Set**. Update and save the configuration.

Configuring Group Grants

Under the group-grant object, you can configure default and attribute grants for specific groups. Group grants apply to users belonging to these groups and matching the resource-identity.

To add a group-grant:

1. Select the **Access** tab and click **authorization**.
2. Click **Add group-grant**.
3. **name**—Enter the name of the group for which you are configuring this grant.
4. Click **Create**. The group-grant object appears.



Configure access\authorization\group-grant GG1

* name	GG1
default-grant	Add default-grant
attribute-grant	Add attribute-grant

5. Click **Add default-grant** to configure a default grant for this group or click **Add attribute-grant** to configure an attribute grant for this group.
6. Configure the default or attribute grant as described above.

   Note: For more information on configuring **default-grants** see Configuring Default Grants. For more information on configuring **attribute-grants** see Configuring Attribute grants.
7. Click **Set**. Update and save the configuration.

There are four new show commands which allow you to view information on your grant configuration: **show authorized-user-privileges**, **show authorized-user-attributes**, **show authorized-user-groups**, and **show authorized-user-summary**.

The **show authorized-user-privileges** action displays information about users' authorization privileges from the user cache.

> **Note:** If a user has never logged into the OS-E, their name does not appear in the cache and, therefore, is not displayed in the **show authorized-user-privileges** command output.

```
NNOS-E>show authorized-user-privileges

username   resource-type privilege identity-type resource-identity
--------   ------------- --------- ------------- -----------------
admin      event-channel C+R+U+D   equals        /system/*
```

Field	Description
username	The name of the configured OS-E user.
resource-type	The resource-type of the grant configured for this user.
privilege	The CRUD privileges of the of the resource-type configured for this user.
identity-type	The method in which the OS-E matches the users' resource-identity.
resource-identity	The value or regular expression the OS-E uses to check users' authorization privileges.

The **show authorized-user-attributes** action displays information about configured OS-E users and their attributes and values.

```
NNOS-E>show authorized-user-attributes

username   attribute                     value
--------   ---------                     -----
sjones     mail                         sjones@acmepacket.com
sjones    msrtcsip-primaryuseraddress   sip:sjones@acmepacket.com
sjones     cn                            Sam Jones
sjones     samaccountname               sjones
sjones     msrtcsip-line                 tel:+17815557256
sjones     st                            MA
sjones     telephonenumber               +1 (781) 555-4839
```

Field	Description
username	The configured OS-E user.
attribute	The attribute name.
value	The value of the attribute for that user.

The **show authorized-user-groups** action displays the configured users and the groups to which they belong from the user cache.

```
NNOS-E>show authorized-user-groups

username     group
--------     -----
sjones       eng
sjones       software
sjones      dev
sjones       ct
sjones      engineering
sjones       deliveries
sjones       funcspec
```

Field	Description
username	The configured OS-E user.
group	The group to which the user belongs.

The **show authorized-user-summary** action displays an abbreviated version of users' authorization privileges from the user cache.

```
NNOS-E>show authorized-user-summary

username    resource-types
--------    --------------
admin      event-channel
test_user   event-channel
```

Field	Description
username	The name of the configured OS-E user.
resource-type	The resource-type of the grant configured for this user.

Configuring Route Headers In an SMX Environment

The OS-E can be integrated into a Net-Net SMX IP Multimedia Subsystem (IMS) environment. The SMX is an integrated session control, policy enforcement, media management, and registration solution that incorporates functional components of the IMS, the Resource and Admission Control Subsystem (RACS), and functions necessary for the interconnection with other IP networks/domains.

In addition, the SMX functions as an S-CSCF in an IMS core. It communicates with the HSS, ENUM infrastructure, or local registration database to obtain Authorization, Authentication, S-CSCF assignment, and ultimately routing instructions. To accomplish these functions, the SMX performs the SIP registrar role in conjunction with an HSS.

When deployed within an SMX environment, you must configure the OS-E to handle route-headers properly. When the OS-E receives a SIP INVITE via the SMX, it must strip off the top route-header and ignore the rest. The OS-E forwards the INVITE back to the SMX by performing a routing calculation based on the configured dial-plan. This calculation uses the request-URI rather than the route-header.

Use the **third-party-call-control > strip-route-header** and **ignore-route-header** properties to configure OS-E route headers.

To configure OS-E route headers in an SMX environment:

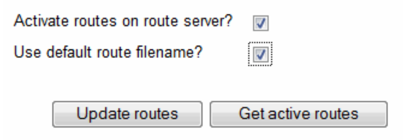
1. Select the **Configuration** tab and click the **vsp** object.
2. Click either the **default-session-config** or **session-config-pool > entry** object.
3. Click **third-party-call-control**.
4. **strip-route-header**—*Secondary property.* Select **top** from the drop-down box.
5. **ignore-route-header**—*Secondary property.* Select **enabled** from the drop-down box.
6. Click **Set**. Update and save the configuration.

Recreating the Route Server Default Route File Name

When you download a route file from the route server to the route server import tool, the route file is named CallRateTableMMM_DD_YYYY_HHMMSS.xml by default where MM_DD_YYYY_HHMMSS is the date and time of the file download.

After initially downloading the file, you may need to send back to the route server for updates. You now have the ability to regenerate the filename upon subsequent downloads to the route server import tool.

A new property has been added to the Route Server page in the route server import tool. When the **Use default route filename** checkbox is checked, the filename is regenerated with the current date and time.



Activate routes on route server? ☑

Use default route filename? ☑

> **Note:** You can only regenerate a route server filename when downloading one file to the route server import tool. The **Use default route filename** property does not appear when you are importing multiple files.

Enabling HTTP Authentication for the Route Server Import Tool

In releases prior to 3.7, the route server import tool sent action requests to the web service using the HTTPS protocol only.

You can now configure the route server import tool to send action requests using HTTP basic authentication. You must first configure protocol and authentication type under the **web-services** object on the route server and then set it to match within the route server import tool.

To enable HTTP authentication on the web service:

1. Under the **Configuration** tab, select the **ip** object under the **cluster > box > interface** object.
2. Select the **web-services** object.
3. **protocol**—Set **type** to **http** and **port** to the web-services port you are using.
4. **authentication**—Set **type** to **basic**.

5. Click **Set**. Update and save the configuration.

To enable HTTP authentication on the route server import tool:

1. Log into the route server import tool.
2. Select the **Route Server** tab.
3. **Web service protocol**—Select **http**. This property specifies the protocol to use for sending requests to the web-service on the route server. Valid values are **http** and **https** and the default setting is **https**.
4. **Remote authentication**—Select **basic**. This property specifies whether a certificate must be sent to the web-service on the route server for certificate authentication or whether the user must specify a username and password to use for authentication. Valid values are **basic** and **certificate** and the default setting is **certificate**.
5. **Remote username**—Enter your web service username.
6. **Remote password**—Enter your web service password.
7. Click either **Update routes** or **Get active routes** to execute the action you want to perform.

Route Server Variable Support

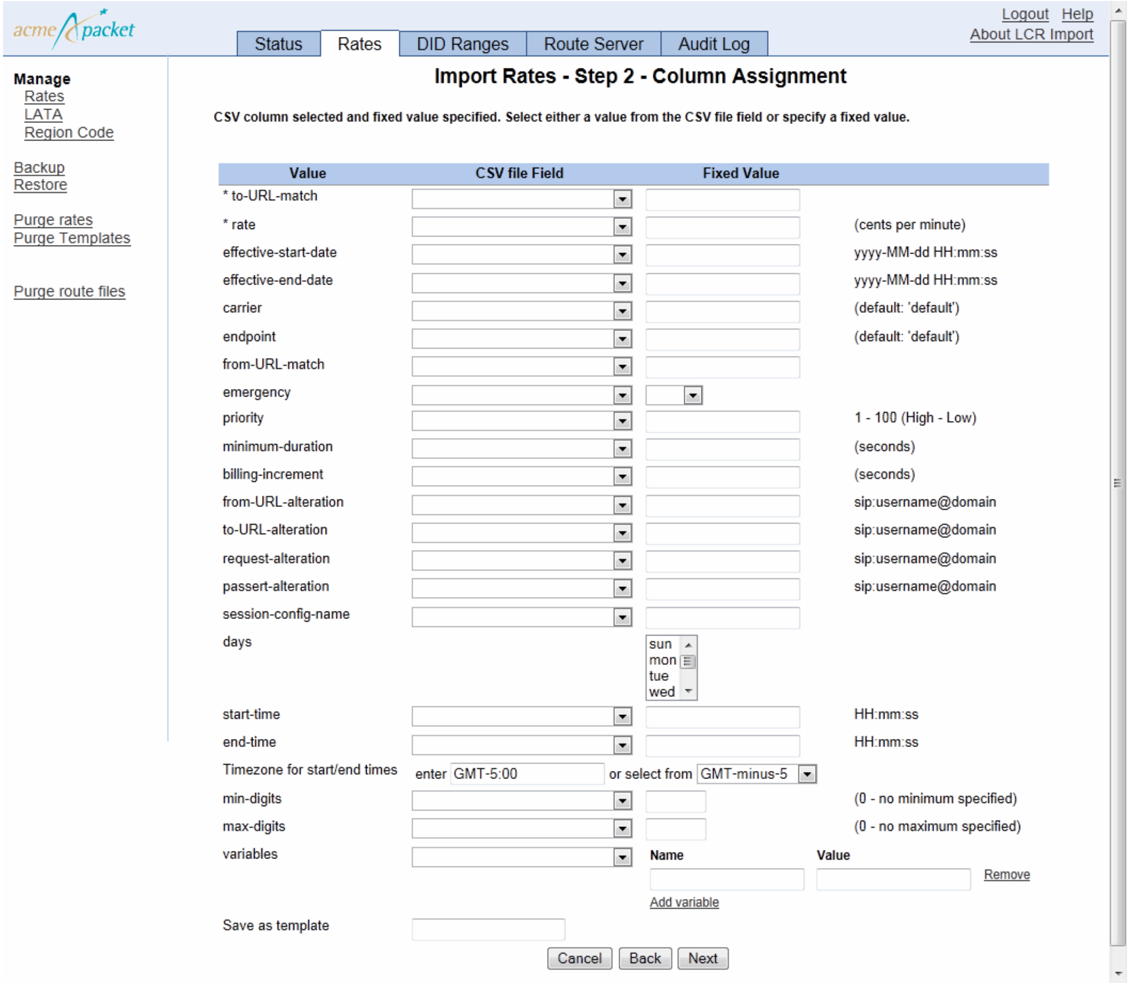
The route server import tool now includes a **variables** field on the Rates and DID Ranges pages. When these values are included, the generated routes in the route files include a variables property.

To add variables to a route file, specify name and value pairs in the CSV file in the format of

name1=value1;name2=value2.

Separate multiple name and value pairs with semi-colons “;”. During Rates and DIDs imports, specify the CSV variables field in the **variables** property in the import tool.

Variable values supersede the CSV file field value if both are specified.



Import Rates - Step 2 - Column Assignment

CSV column selected and fixed value specified. Select either a value from the CSV file field or specify a fixed value.

Value	CSV file Field	Fixed Value	
* to-URL-match			
* rate			(cents per minute)
effective-start-date			yyyy-MM-dd HH:mm:ss
effective-end-date			yyyy-MM-dd HH:mm:ss
carrier			(default: 'default')
endpoint			(default: 'default')
from-URL-match			
emergency			
priority			1 - 100 (High - Low)
minimum-duration			(seconds)
billing-increment			(seconds)
from-URL-alteration			sip:username@domain
to-URL-alteration			sip:username@domain
request-alteration			sip:username@domain
passert-alteration			sip:username@domain
session-config-name			
days		sun mon tue wed	
start-time			HH:mm:ss
end-time			HH:mm:ss
Timezone for start/end times	enter GMT-5:00	or select from GMT-minus-5	
min-digits			(0 - no minimum specified)
max-digits			(0 - no maximum specified)
variables		Name	Value
			Remove
		Add variable	
Save as template			

Upgraded 64-Bit Kernel

The OS-E now runs on a 64-bit kernel to support compatibility with a broader range of servers.

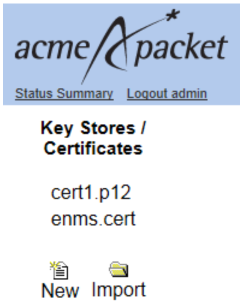
For more information on installing the OS-E as a VM, see the Oracle Communications OS-E Installation and Commissioning Guide.

Supporting Subject Alternative Name for HTTPS Certificates

The OS-E now supports Subject Alternative Name (SAN) for use with HTTPS certificates. SAN is a X509 version 3 certificate extension that allows one to specify a list of host names protected by a single SSL certificate.

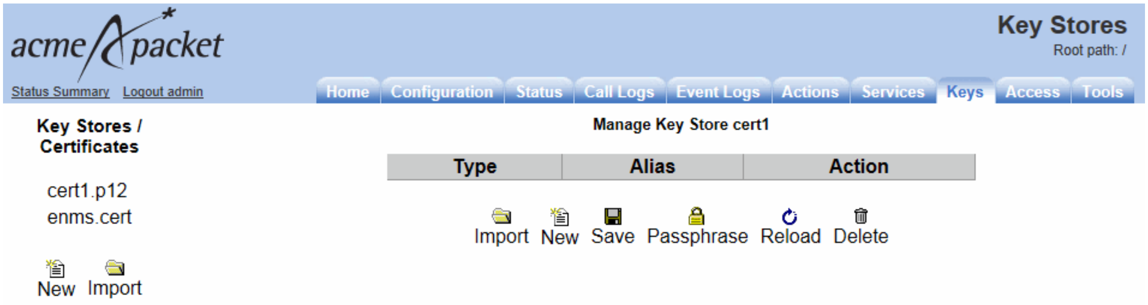
To add multiple SANs to a certificate:

1. Select the **Keys** tab and either click **New** to create a new key store or select the existing store on which you want to add a certificate.



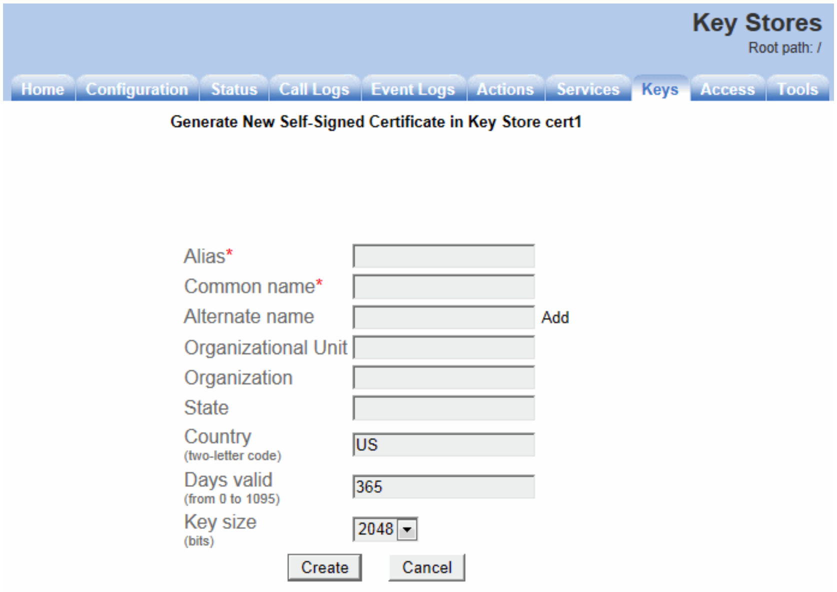
2. Enter a name and passphrase if creating a new keystore and click **Create**.

   The key store appears.



3. Click **New**. The Generate New Self-Signed Certificate in Key Store *X* page appears.

4. Click **Add** beside the **Alternate name** field to add alternate host names be added to the certificate’s subjectAltName field.



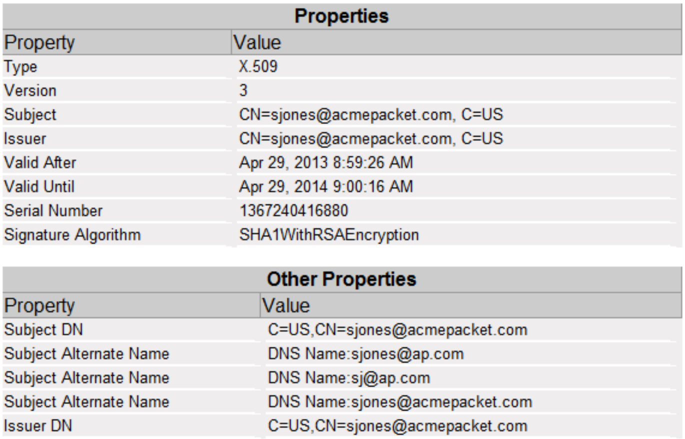
Generate New Self-Signed Certificate in Key Store cert1

Alias*	
Common name*	
Alternate name	
Organizational Unit	
Organization	
State	
Country (two-letter code)	US
Days valid (from 0 to 1095)	365
Key size (bits)	2048

5. Click **Create**. The certificate appears in the key store.

   Note: When configuring the OS-E via the CLI, separate multiple SAN entries using the ‘|’ character.

To view the SANs within a certificate click **View** next to the certificate name. The following image shows three SANs.



Properties

Property	Value
Type	X.509
Version	3
Subject	CN=sjones@acmepacket.com, C=US
Issuer	CN=sjones@acmepacket.com, C=US
Valid After	Apr 29, 2013 8:59:26 AM
Valid Until	Apr 29, 2014 9:00:16 AM
Serial Number	1367240416880
Signature Algorithm	SHA1WithRSAEncryption

Other Properties

Property	Value
Subject DN	C=US,CN=sjones@acmepacket.com
Subject Alternate Name	DNS Name:sjones@ap.com
Subject Alternate Name	DNS Name:sj@ap.com
Subject Alternate Name	DNS Name:sjones@acmepacket.com
Issuer DN	C=US,CN=sjones@acmepacket.com

Custom Scheme Support in SIP Headers

The OS-E now supports URL schemes other than sip, sips, and tel to be proxied through the OS-E in SIP message headers.

There are two ways to add allowed custom URL schemes. The **vsp > settings > sip-allow-schemes** property has been created to add the URL schemes allowed to be proxied through.

You can also add custom schemes via the **session-config > header-settings > allowed-header** property.

SIP Proxy Enhancements

A new set of OS-E actions, **sip-send-** have been created which allow external applications to send out-of-dialog SIP messages.

The **sip-send-notify** command sends out-of-dialog SIP NOTIFY requests. The action syntax is:

```
sip-send-notify <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the subscription binding (returned in SubscribeEvent).
- [*session-config*]—The session-config used for formatting NOTIFY headers and bodies.

The **sip-send-subscribe** action sends an out-of-dialog SIP SUBSCRIBE request.

When an application either sends a SUBSCRIBE request and receives a 200 OK or accepts a SUBSCRIBE request by responding with a 200 OK, the OS-E creates a subscription binding. This binding contains the supported events and a unique ID. The application can use this ID to distinguish between subscriptions that it created and other subscriptions created for the same AOR but a different application.

Subscription bindings are arranged in a vector under the AOR and have an expiration time that can be specified in the **sip-send-subscribe** and **call-control-send-subscribe** actions. If you do not specify an expiration time, the default is 3600 seconds.

The action syntax is:

```
sip-send-subscribe <AOR> [event] [accept] [id] [session-config]
   [expiration]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*event*]—The name of the event type to subscribe to.
- [*accept*]—The acceptable event data to include.
- [*id*]—The unique ID that identifies the subscription binding (returned in SubscribeEvent).
- [*session-config*]—The session-config used for formatting SUBSCRIBE headers and bodies.
- [*expiration*]—The expiration time of the binding, in seconds. If this value is not specified, the registration-plan expiration is used.

The **sip-send-options** action sends an out-of-dialog SIP OPTIONS request. The action syntax is:

```
sip-send-options <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the registration binding (returned in RegisterEvent).
- [*session-config*]—The session-config used for formatting OPTIONS headers and bodies.

The **sip-send-message** action sends out-of-dialog SIP MESSAGE requests. The action syntax is:

```
sip-send-options <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the registration binding (returned in RegisterEvent).

- [*session-config*]—The session-config used for formatting MESSAGE headers and bodies.

The **sip-send-info** action sends out-of-dialog SIP INFO requests. The action syntax is:

```
sip-send-info <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the registration binding (returned in RegisterEvent).
- [*session-config*]—The session-config used for formatting INFO headers and bodies.

The **sip-send-other** action sends out-of-dialog requests for all other SIP methods which do not have a specific action. The action syntax is:

```
sip-send-other <method> <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*method*>—The SIP method to use in the request.
- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the registration binding (returned in RegisterEvent).
- [*session-config*]—The session-config used for formatting the request headers and bodies.

In addition, a new set of **call-control-send-** actions have been created which allow external applications to send IP requests inside an INVITE dialog.

The **call-control-send-notify** action sends an in-dialog SIP NOTIFY request. The action syntax is:

```
call-control-send-notify <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the subscription binding (returned in SubscribeEvent).

- [*session-config*]—The session-config used for formatting NOTIFY headers and bodies.

The **call-control-send-subscribe** action sends an in-dialog SIP SUBSCRIBE request.

When an application either sends a SUBSCRIBE request and receives a 200 OK or accepts a SUBSCRIBE request by responding with a 200 OK, the OS-E creates a subscription binding. This binding contains the supported events and a unique ID. The application can use this ID to distinguish between subscriptions that it created and other subscriptions created for the same AOR but a different application.

Subscription bindings are arranged in a vector under the AOR and have an expiration time that can be specified in the **sip-send-subscribe** and **call-control-send-subscribe** actions. If you do not specify an expiration time, the default is 3600 seconds.

The action syntax is:

```
call-control-send-subscribe <AOR> [event] [accept] [id]
   [session-config] [expiration]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*event*]—The name of the event type to subscribe to.
- [*accept*]—The acceptable event data to include.
- [*id*]—The unique ID that identifies the subscription binding (returned in SubscribeEvent). This value is optional on a first call but required on subsequent re-subscribes.
- [*session-config*]—The session-config used for formatting SUBSCRIBE headers and bodies.
- [*expiration*]—The expiration time of the binding, in seconds. If this value is not specified, the default is **3600** seconds.

The **call-control-send-options** action sends in-dialog SIP OPTIONS requests. The action syntax is:

```
call-control-send-options <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.

- [*id*]—The Unique ID that identifies the registration binding (returned in SubscribeEvent).
- [*session-config*]—The session-config used for formatting OPTIONS headers and bodies.

The **call-control-send-message** sends an in-dialog SIP MESSAGE request. The action syntax is:

```
call-control-send-messages <AOR> <from> [id] [content-type] [body]
   [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- <*from*>—The originating SIP URI of the call.
- [*id*]—The Unique ID that identifies the registration binding (returned in SubscribeEvent).
- [*content-type*]—Should be set to **text/plain.**
- [*body*]—The content of the message.
- [*session-config*]—The session-config used for formatting MESSAGE headers and bodies.

The **call-control-send-info** action sends an in-dialog SIP INFO request. The action syntax is:

```
call-control-send-info <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the registration binding (returned in SubscribeEvent).
- [*session-config*]—The session-config used for formatting INFO headers and bodies.

The **call-control-send-other** action sends in-dialog requests for all other SIP methods which do not have a specific action. The action syntax is:

```
call-control-send-other <method> <AOR> [id] [session-config]
```

Valid arguments for this action are:

- <*method*>—The SIP method to use in the request.
- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the registration binding (returned in RegisterEvent).
- [*session-config*]—The session-config used for formatting the request headers and bodies.

The **subscribe-flush** action removes all subscription bindings from the entire cluster. The action syntax is:

```
subscribe-flush
```

There are no arguments for this action.

The **subscribe-delete** action removes either a single subscription binding or all of the bindings for a single AOR. The action syntax is:

```
subscribe-delete <AOR> [id]
```

Valid arguments for this action are:

- <*AOR*>—The Address of Record in the form of a URI.
- [*id*]—The Unique ID that identifies the subscription binding (if not specified, all bindings are removed).

The OS-E now supports involving external applications in the processing of SIP requests. When enabled on a registration-plan, the OS-E returns additional structured data in the SIP message event that notifies the external application that SIP processing is pending and it needs to do something. In order for SIP processing to resume, the external application must reply back with either a **proxy-accept**, **proxy-discard**, or **proxy-reject**.

The **proxy-accept** action tells the proxy state machine that the request referenced by the handle (returned in the SIP message event) can proceed. You can also specify the response code, text string, and session-config reference if additional header or body manipulation is required on the response.

The action syntax is:

```
proxy-accept <handle> [response-code] [response-string]
   [session-config]
```

Valid arguments for this action are:

- <*handle*>—References the proxy session handling the request (this value is returned in the SIP message event).
- [*response-code*]—The SIP response code to return in response. By default the OS-E uses the response-code configured in the registration-plan.
- [*response-string*]—The SIP response string to return in the response. By default the OS-E uses the response-string configured in the registration-plan.
- [*session-config*]—The session-config to use for formatting the response headers and bodies.

The **proxy-reject** action tells the proxy state machine that the request referenced by the handle (returned in the SIP message event) has been rejected. You can also specify the response code, text string, and session-config reference if additional header or body manipulation is required on the response.

The action syntax is:

```
proxy-reject <handle> [response-code] [response-string]
   [session-config]
```

Valid arguments for this action are:

- <*handle*>—References the proxy session handling the request (this value is returned in the SIP message event).
- [*response-code*]—The SIP response code to return in response. By default the OS-E uses the response-code configured in the registration-plan.
- [*response-string*]—The SIP response string to return in the response. By default the OS-E uses the response-string configured in the registration-plan.
- [*session-config*]—The session-config to use for formatting the response headers and bodies.

The **proxy-discard** action tells the proxy state machine that the request referenced by the handle (returned in the SIP message event) has been rejected and to silently discard the message.

The action syntax is:

```
proxy-discard <handle>
```

Valid arguments for this action are:

- <*handle*>—References the proxy session handling the request (this value is returned in the SIP message).

Whitelist and Blacklist Permissions Filtering

In releases prior to 3.7, there was one way to filter actions which users had access to specific actions, via the **access > permissions > action-filter** property.

You can now create an **action-filter-blacklist** and an **action-filter-whitelist**.

Via the **action-filter-blacklist** property, you can select an action-filter containing a list of actions you want to restrict certain users from using. When a user attempts to execute a restricted action, he gets the following error message:

```
Insufficient permissions for user
```

Via the **action-filter-whitelist** property, you can select an action-filter containing a list of actions you want to allow certain users to use.

The action-filter-whitelist property supports the use of a wildcard. The wildcard is an asterisk (*) that can be located at the end of a string only. For example, to create an action-filter for all call-control actions, enter **call-control-***.

When action-filters are configured on the OS-E, the OS-E always checks the **action-filter-blacklist** settings first. If the action is found on the blacklist, the user is not allowed to use it.

If both the **action-filter-blacklist** and **action-filter-whitelist** are configured and an action does not appear on either list, the user is restricted from using the action.

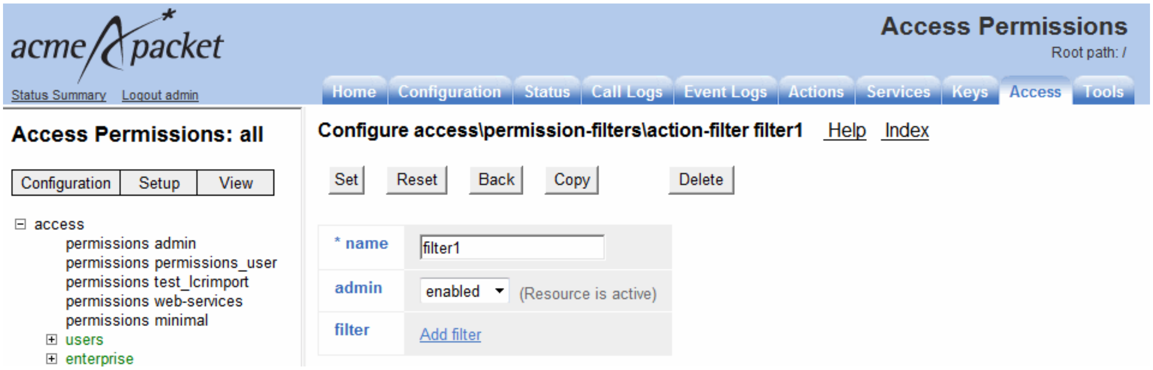
If an action is not found on the **action-filter-blacklist** and **action-filter-whitelist** is not configured, the user is allowed to use it.

> **Note:** You must enter actions into the **action-filter-blacklist** and **action-filter-whitelist** properties without any arguments. When anything more than an action name is specified, the OS-E ignores the filter.

To configure an **action-filter**:

1. Select the **Access** tab and click **Access**.
2. Click **Configure** next to **permission-filters**.
3. Click **Add action-filter**.

4. **name**—Specify a name to give this **action-filter**.
5. Click **Create**. The **filter** object appears.



Configure access\permission-filters\action-filter filter1

* name	filter1
admin	enabled (Resource is active)
filter	Add filter

6. **admin**—Set to **enabled** to enable this action-filter.
7. Click **Add filter**.
8. **filter**—Specify an action, without any arguments, to be applied to this filter.

   Note: If you enter an action with arguments, the action is ignored.
9. Repeat Steps 7 and 8 for as many actions you want to apply to this filter.
10. Click **Set**. Update and save the configuration.

Once you have created config-filters and action-filters, you must apply them to a permission set.

To apply config-filters and action-filters to a permissions set:

1. Select the **Access** tab and click **Access**.
2. Click **Add permissions** to create a new permissions set or click **Edit** next to an existing permissions set.
3. **config-filter**—Select a config-filter from the drop-down list whose configuration objects you want to restrict users with this permissions set from using. If you have not yet created a config-filter, click Create next to this property.
4. **action-filter-blacklist**—Select an action-filter from the drop-down list whose actions you want to restrict users with this permissions set from using. If you have not yet created an action-filter, click Create next to this property.

5. **action-filter-whitelist**—Select an action-filter from the drop-down list whose actions you want to allow users with this permissions set to use. If you have not yet created an action-filter, click Create next to this property.
6. Click **Set**. Update and save the configuration.

For more information on configuring access, permissions sets, and users see the Oracle Communications OS-E System Administration Guide.

Mitigating Application-Level DoS Attacks on OS-E HTTP Services

The OS-E now supports application-layer DoS attack defense for OS-E processes exposed to the web to help prevent application crashing and minimizing resource depletion.

DoS policies prior to release 3.7 protect general OS-E resources, while the new policies protect the web-services and web management interface, the eventpush process (cometD application), and the websocket processes. These policies provide the following defenses:

- Limits the number of connections
- Limits the number request processing threads
- Limits the number of sessions
- Rejects un-authenticated requests
- Times-out idle sessions

As part of these DoS enhancements, the OS-E now supports Cross-Origin Resource Sharing (CORS). CORS is a web browser technology that allows JavaScript on a webpage to make HTTP requests to a different domain than the JavaScript originated from. CORS defines a way in which the browser and server interact to determine whether or not to allow the cross-origin request.

> **Note:** Both the server and browser must support CORS for it to work.

The OS-E supports CORS for both the web-services and eventpush-service processes.

In release 3.7, you can configure the web-services and eventpush-service CORS support to do one of three things:

- Allow CORS requests from any origin
- Allow CORS requests for specified origins only

- Do not allow any CORS requests

Several new **web-service** configuration properties have been created to help mitigate DoS for web-services and the web management interface:

- **max-sessions**—*Advanced property.* Specifies the maximum number of concurrent sessions allowed. The default value is **10000**. The valid range is an integer from **0** (there is no limit) to **200000**.
- **accept-connections**—*Advanced property.* Specifies the number of connection requests that can be queued up before the OS-E sends rejections. The default value is **100**. The valid range is an integer from **0** to **200000**.
- **max-connections**—*Advanced property.* The maximum number of connections the server can accept and process at any given time. When this threshold is reached, the server does not accept any more connections until the number falls below this value. The default value is **10000**. The valid range is an integer from **0** to **200000**.

  Note: Based on the **accept-connections** value, the OS-E may still accept connections into the queue.
- **idle-connection-timeout**—*Advanced property.* Specifies the maximum time a connection can idle before it times out and the OS-E drops it. The default value is **20**. The valid range is an integer from **0** to **200000**.
- **session-idle-timeout**—Specifies the maximum time in seconds that this web-service's sessions can idle before it times out and the OS-E drops it. The default value is **30**. The valid range is an integer from **0** to **4294967296**.
- **max-keep-alive-requests**—*Advanced property.* Specifies the maximum number of HTTP requests which can be queued before the connection is closed by the server. The default value is **-1**, meaning the OS-E allows an unlimited number of queued or keep-alive HTTP requests. The valid range is an integer from **-1** to **200000**.
- **cross-origin-resource-sharing**—*Advanced property.* Access the web-services CORS settings.
  - **enable-cross-origin**—*Advanced property.* Enables or disables cross-origin requests to the web-server. The default value is **accept-any-origins**. The valid values are **accept-any-origins**, **accept-specified-origins**, and **deny-all-origins**.

  - **allowed-cross-origin**—*Advanced property.* Specifies the origins whose requests the OS-E accepts when the **enable-cross-origin** property is set to **accept-specified-origins**. There is no default setting.

Several new **eventpush-service** configuration properties have been created to help mitigate DoS for the CometD application:

- **max-sessions**—*Advanced property.* Specifies the maximum number of concurrent sessions allowed. The default value is **10000**. The valid range is an integer from **0** (there is no limit) to **200000**.
- **accept-connections**—*Advanced property.* Specifies the number of connection requests that can be queued up before the OS-E sends rejections. The default value is **100**. The valid range is an integer from **0** to **200000**.
- **idle-connection-timeout**—*Advanced property.* Specifies the maximum time a connection can idle before it times out and the OS-E drops it. The default value is **20**. The valid range is an integer from **0** to **200000**.
- **session-idle-timeout**—Specifies the maximum time a session can idle before it times out and the OS-E drops it. The default value is **60**. The valid range is an integer from **0** to **4294967296**.
- **cross-origin-resource-sharing**—*Advanced property.* Access the eventpush-service CORS settings.
  - **enable-cross-origin**—*Advanced property.* Enables or disables cross-origin requests to the web-server. The default value is **accept-any-origins**. The valid values are **accept-any-origins**, **accept-specified-origins**, and **deny-all-origins**.
  - **allowed-cross-origin**—*Advanced property.* Specifies the origins whose requests the OS-E accepts when the **enable-cross-origin** property is set to **accept-specified-origins**. There is no default setting.

When the OS-E implements SIP over WebSockets, it uses an HTTP handshake to initiate the session before making the transition to SIP over WebSockets.

The OS-E uses a timer during the HTTP handshake. If an incoming connection does not successfully transition to SIP over WebSockets in this internally set time, the connection is abandoned and the event is reported to the DoS system.

You can configure the OS-E to authenticate remote clients during the HTTP handshake phase. The **ws-port > http-authentication** property has been created to allow you to specify the type of authentication mode to use for the HTTP handshake. The default setting is **none**. Valid values are:

- none—No HTTP authentication for WebSockets
- local—Local HTTP authentication for WebSockets
- radius—RADIUS HTTP authentication for WebSockets
- diameter—Diameter HTTP authentication for WebSockets
- directory—Directory HTTP authentication for WebSockets (this references the **vsp** > **enterprise** > **directories** configuration)
- accept—Accept all HTTP requests for WebSockets
- reject—Reject all HTTP requests for WebSockets

When authentication is enabled, the OS-E responds to a client's initial GET with a 401 Unauthorized. If the client responds with valid credentials, the connection proceeds. If the client does not respond with valid credentials, the connection is dropped and the event is reported to the DoS system for further policy-based action.

In addition, WebSocket Secure (WSS) connections use TLS and you can configure the OS-E to require the remote client to present a certificate for authentication via the **tls** > **certificate** > **peer-certificate-verification** and **specific-ca-file** properties. For more information on these properties, see the Oracle Communications OS-E Objects and Properties Reference Guide.

You can limit the number of active connections on WebSockets by using the existing **vsp** > **settings** > **sockets-per-box-max** and **sockets-per-peer-max** properties. For more information on these properties, see the Oracle Communications OS-E Objects and Properties Reference Guide.

Several new show status commands have been created to display information about your DoS mitigation configuration settings and activity.

The **show web-services-status** command displays information about the **web-services** configuration and activity on the OS-E.

```
NNOS-E>show web-services-status

                           ip: 100.40.10.7
                         port: 8082
```

```
                     sessions: 0
                 max-sessions: 10000
         max-sessions-reached: 0
         session-idle-timeout: 30 minutes
                 pool-threads: 0
                  max-threads: 10
                  connections: 0
              max-connections: 10000
      idle-connection-timeout: 20 seconds
      keep-alive-requests-max: -1
 cross-origin-requests-denied: 0
cross-origin-requests-allowed: 0
```

Field	Description
ip	The web-service IP address.
port	The web-service port number.
sessions	The number of active sessions on this web-service.
max-sessions	The configured maximum number of sessions allowed.
max-sessions-reached	The number of times a session was not created because the **max-sessions** value was reached.
session-idle-timeout	The configured session idle timeout.
pool-threads	The current number of request processing threads in the thread pool.
max-threads	The configured maximum number of request processing threads.
connections	The current number of connections.
max-connections	The configured maximum number of connections allowed.
idle-connection-timeout	The configured connection idle timeout.
keep-alive-requests-max	The configured maximum number of HTTP requests which can be queued before the connection is closed by the server.
cross-origin-requests-denied	The number of CORS requests allowed.
cross-origin-requests-allowed	The number of CORS requests denied.

The **show cometd-status** command displays information about the **eventpush-service** configuration and activity on the OS-E.

```
NNOS-E>show cometd-status

                        ip: 100.40.10.7
                      port: 8081
                  sessions: 0
              max-sessions: 10000
      max-sessions-reached: 0
      session-idle-timeout: 60 seconds
              pool-threads: 1
               max-threads: 10
   idle-connection-timeout: 20 seconds
               connections: 0
 cross-origin-requests-denied: 0
cross-origin-requests-allowed: 0
```

Field	Description
ip	The eventpush-service IP address.
port	The eventpush-service port number.
sessions	The number of active sessions.
max-sessions	The configured maximum number of sessions allowed.
max-sessions-reached	The number of times a session was not created because the **max-sessions** value was reached.
session-idle-timeout	The configured session idle timeout.
pool-threads	The current number of request processing threads in the thread pool.
max-threads	The configured maximum number of request processing threads.
idle-connection-timeout	The configured connection idle timeout.
connections	The current number of connections.
cross-origin-requests-denied	The number of CORS requests allowed.
cross-origin-requests-allowed	The number of CORS requests denied.

The **show web-ext-status** command displays information about the web management interface configuration and activity.

```
NNOS-E>show web-ext-status

                  ip: 100.40.10.7
                port: 80
            sessions: 0
        max-sessions: 30
```

```
   max-sessions-reached: 0
   session-idle-timeout: 30 minutes
           pool-threads: 1
            max-threads: 10
            connections: 0
        max-connections: 10000
idle-connection-timeout: 20 seconds
keep-alive-requests-max: -1
```

Field	Description
ip	The web management interface IP address.
port	The web management interface port number.
sessions	The number of active sessions.
max-sessions	The configured maximum number of sessions allowed.
max-sessions-reached	The number of times a session was not created because the **max-sessions** value was reached.
session-idle-timeout	The configured session idle timeout.
pool-threads	The current number of request processing threads in the thread pool.
max-threads	The configured maximum number of request processing threads.
connections	The current number of connections.
max-connections	The configured maximum number of connections allowed.
idle-connection-timeout	The configured connection idle timeout.
keep-alive-requests-max	The configured maximum number of HTTP requests which can be queued before the connection is closed by the server.

The **show ws-listener** command displays information about the configured SIP over WebSocket listener port.

```
NNOS-E>show ws-listener
---------------------------------------------------------------------
Process Intf Port Attempts Success Rejects Failed In-Flight Current
--------------------------------------------------------------------
SIP     eth0 9080      5       5       0      0       0       0
--------------------------------------------------------------------
```

Field	Description
Process	The process using this WS listener port.
Intf	The interface on which this WS listener port is configured.
Port	The port number on which this WS listener is configured.
Attempts	The number of times a remote client has attempted to reach this WS listener port.
Success	The number of times a remote client was successful in their attempt to reach this WS listener port.
Rejects	The number of times the OS-E rejected a remote client trying to access this WS listener port.
Failed	The number of times a remote client failed to access this WS listener port.
In-Flight	The number of in-flight sessions currently trying to access this WS listener port.
Current	The current number of sessions on this WS listener port.

The **show wss-listener** command displays information about the configured SIP over WebSocket Secure (WSS) listener port.

```
NNOS-E>show wss-listener
---------------------------------------------------------------------
Process Intf Port Attempts Success Rejects Failed In-Flight Current
--------------------------------------------------------------------
SIP     eth0 9090      5       5       0      0       0       0
--------------------------------------------------------------------
```

Field	Description
Process	The process using this WSS listener port.
Intf	The interface on which this WSS listener port is configured.
Port	The port number on which this WSS listener is configured.
Attempts	The number of times a remote client has attempted to reach this WS listener port.

Field	Description
Success	The number of times a remote client was successful in their attempt to reach this WSS listener port.
Rejects	The number of times the OS-E rejected a remote client trying to access this WS listener port.
Failed	The number of times a remote client failed to access this WSS listener port.
In-Flight	The number of in-flight sessions currently trying to access this WSS listener port.
Current	The current number of sessions on this WSS listener port.

Call-Control Action Updates

The **call-control <*argument*>** actions have been deprecated. They have been replaced by the following **call-control-<*argument*>** actions:

call-control-accept

Accepts an incoming call from an offering endpoint.

> **Note:** You must specify content-type as application/sdp and body as the SDP for the call.

Syntax

```
call-control-accept <handle> [content-type] [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-annotate

Annotates the text you specify to a call leg.

Syntax

```
call-control-annotate <handle> <text>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*text*>—The annotated text you are providing to the call leg.

call-control-attach

Attaches a call leg to an existing SIP session.

Syntax

```
call-control-attach <handle> <session-id>
```

Arguments

- <*handle*>—The handle of the endpoint to be attached.
- <*session-id*>—The session to which the endpoint is being attached.

call-control-call

Initiates a call using provided To and From SIP URIs.

You can configure the OS-E to add post-dial digits to a call-control-call action by appending the string **postd=*digits*** to the user portion of the **To** argument.

Syntax

```
call-control-call <to> <from> [requestId] [originatorFirst]
[async] [transport] [config] [session-id] [content-type] [body]
```

Arguments

- <*to*>—The destination SIP URI of the session.
- <*from*>—The originating SIP URI of the session.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.

- [*originatorFirst*]—When **enabled** (the default), the originating party is connected first. When **disabled**, the called party is connected first.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*transport*]—The transport method to use for the call. This can be set to **any**, **TCP**, **UDP**, or **TLS**.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

- Enclose the value in quotation marks when using the CLI.
- [*session-id*]—The optional session ID for the session.
- [*content-type*]—The content type of the message body of the initial call.
- [*body*]—The message body of the initial call.

call-control-call-to-session

Initiates a call to an existing session.

Syntax

```
call-control-call-to-session <to> <from> <session-id> [requestId]
   [originatorFirst] [async] [transport] [config] [content-type]
   [body]
```

Arguments

- <*to*>—The destination SIP URI of the session.
- <*from*>—The originating SIP URI of the session.
- <*session-id*>—The optional session ID for the session.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*originatorFirst*]—When **enabled** (the default), the originating party is connected first. When **disabled**, the called party is connected first.

- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*transport*]—The transport method to use for the call. This can be set to **any**, **TCP**, **UDP**, or **TLS**.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-connect

Connects an existing parked call leg to a given endpoint. If the called party ends the call, the original call reverts back to a parked state.

Syntax

```
call-control-connect <handle> <endpoint> [async] [requestId] [park]
   [config]
```

Arguments

- <*handle*>—Identifies the leg of a call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*endpoint*>—The URI of the call's destination.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*park*]—When enabled, the outgoing call leg persists and reverts to a parked state when its peer is terminated.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

call-control-create-session

Creates a rendezvous session to which you can then add call-legs, add named-variables, or destroy the session. The OS-E automatically assigns the session a unique 64-bit session ID.

Syntax

```
call-control-create-session [requestId] [to] [from]
```

Arguments

- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*to*]—The To URI for the rendezvous session.
- [*from*]—The From URI for the rendezvous session.

call-control-custom

Creates and controls established calls and overrides specific session configuration settings.

Syntax

```
call-control-custom <call> <to> <from> [requestId] [originatorFirst]
   [async] [transport] [config] [session-id]
```

Arguments

- <*call*>—Initiates a call using provided To and From SIP URIs.
- <*to*>—The To URI for the session.
- <*from*>—The From URI from the session.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*originatorFirst*]—When **enabled** (the default), the originating party is connected first. When **disabled**, the called party is connected first.

- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*transport*]—The transport method to use for the call. This can be set to **any**, **TCP**, **UDP**, or **TLS**.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

Enclose the value in quotation marks when using the CLI.

- [*session-id*]—The optional session ID for the session.

call-control-destroy-session

Destroys a rendezvous session.

Syntax

```
call-control-destroy-session [session-id]
```

Arguments

- [*session-id*]—The session-id for the rendezvous session you are destroying. This is the unique 64-bit session ID given to the session by the OS-E when it was created.

call-control-detach

Detaches a call leg from an existing SIP session. If you do not specify a session ID, the OS-E creates a new parked session with that call leg. If you specify a session ID, the OS-E parks the call leg to that existing session.

Syntax

```
call-control-detach <handle> [session-id]
```

Arguments

- <*handle*>—The handle of the endpoint from which you are detaching.
- [*session-id*]—The optional session ID to which you are parking this call leg. If you do not specify a session ID, the OS-E creates a new session.

call-control-detach-to-session

Detaches a call leg and parks it to an existing specified session.

Syntax

```
call-control-detach-to-session <handle> <session-id>
```

Arguments

- <*handle*>—The handle of the endpoint from which you are detaching.
- <*session-id*>—The session ID to which you are parking this call leg.

call-control-disconnect

Terminates all parties in a SIP session.

Syntax

```
call-control-disconnect <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-drop-file

Plays the specified audio file to the party connected to the call leg. When finished, the ASC terminates the call leg.

Syntax

```
call-control-drop-file <handle> <filename> [async]
```

Arguments

- <*handle*>—Identifies the leg of a call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*filename*>—The name of the audio file where a message is recorded or from where a message is played. Audio files must be .wav files in 44.1 kHz, 16-bit mono PCM format. If you give an invalid filename, it is placed in or taken from the /cxc directory.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.

call-control-fork

Adds a new endpoint’s SIP URI to the parked call. The endpoint can receive media but cannot send it. Multiple endpoints can be added using this action.

Syntax

```
call-control-fork <handle> <endpoint> [async] [config]
```

Arguments

- <*handle*>—Identifies the leg of a call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*endpoint*>—The URI of the call’s destination.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

call-control-get-annotation

Retrieves the annotated text given to the call leg.

Syntax

```
call-control-get-annotation <handle>
```

Arguments

- <*handle*>—Identifies the leg of a call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-hold

Places the specified call leg on hold. This puts the media of that call leg into send-only mode. The media of the other call leg, if present, is put into receive-only mode.

Syntax

```
call-control-hold <handle>
```

Arguments

- <*handle*>—Identifies the leg of a call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-info-request

Sends an INFO on an existing call.

Syntax

```
call-control-info-request <handle> [info-package] [content-type]
   [body]
```

Arguments

- <*handle*>—Identifies the leg of a call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*info-package*]—The INFO message to send to the existing call.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-insert-dtmf

Inserts DTMF digits into SIP sessions. DTMF is inserted only into the call leg specified; the other party does not hear it.

Syntax

```
call-control-insert-dtmf <handle> <digits> [volume] [duration]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*digits*>—Specifies the digits inserted into the call leg.
- [*volume*]—The volume of the DTMF digits, in decimals from -36 to 0. The value **1** is the default.
- [*duration*]—The duration of each digit in milliseconds, from 100 to 10000. The value **0** is the default.

call-control-intercept

Connects an incoming call to an existing parked call.

Syntax

```
call-control-intercept <handle> <target>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*target*>—The handle of the parked call.

call-control-join

Connects the parties of two separate calls together. The original call legs, identified by handle1 and handle2, are disconnected.

Syntax

```
call-control-join <handle1> <handle2>
```

Arguments

- <*handle1*>—Identifies the leg of the first call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*handle2*>—Identifies the leg of the second call. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-media-pause

Pauses the playing of an audio file on an active call leg.

Syntax

```
call-control-handle-pause <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-media-resume

Resumes the playing of an audio file on an active call leg.

Syntax

```
call-control-media-resume <handle>
```

Arguments

<*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-media-scanner-start

Attaches a media scanner to a call-leg (in-leg or out-leg) and begins analyzing the signal strength of the received audio. The media scanner reports events when the signal strengths detected cross from the low-threshold property settings to the high-threshold property settings, or vice-versa, and based on the configuration for the media-scanner settings. The media-scanner configuration is retrieved from the session-config associated with the target call. You can specify a named session config, which overrides the session config. The media-scanner settings configuration applied is based on the following precedence:

- **in-media-scanner-settings**—media scanner settings per in-leg call
- **out-media-scanner-settings**—out media scanner settings per out-leg call
- **session-config-media-scanner-settings**—media scanner settings per session-config
- **default-media-scanner-settings**—default property settings

The media scanner will report one of the following events when a transition has occurred:

- **Short-pause**—When a transition (for example, from stable tone to quiet) takes less time than the low-long-duration property setting, such as less than 200 milliseconds
- **Long-pause**—When a transition (for example, from stable tone to quiet) takes more time than the low-long-duration property setting, such as more than 200 milliseconds

- **Short-talk**—When the media-scanner detects talk less than the high-long-duration property setting, such as less than 900 milliseconds
- **Long-talk**—When the media-scanner detects talk longer than the high-long-duration property setting, such as longer than 900 milliseconds
- **Stable-tone**—When the media-scanner detects a constant signal over a sample interval as determined by the averaging window.

Syntax

```
call-control-media-scanner-start <handle> [config]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

call-control-media-scanner-stop

Detaches a media scanner from a call leg.

Syntax

```
call-control-media-scanner-stop <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-media-seek

Seeks a specific point in a monitored recording file.

Syntax

```
call-control-media-seek <handle> <seek-offset> [position]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*seek-offset*>—The offset, in milliseconds, to begin seeking. A negative value seeks backwards. Seeking starts at the spot specified by the **position** parameter.
- [*position*]—Indicates the position to begin seeking:
  - start—Seek from the start of the file. This is the default behavior.
  - current—Seek from the current position of the file.
  - end—Seek from the end of the file.

call-control-media-stop

Stops the playing of an audio file on an active call leg.

Syntax

```
call-control-media-stop <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-memo-begin

Records a message from the parked party, identified by a call leg handle, and stores it in a file you specify.

> **Note:** When **cluster** is **enabled**, **master-service > file-mirror** must be enabled for it to work properly

Syntax

```
call-control-memo-begin <handle> <filename> [greeting] [cluster]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

- <*filename*>—The name of the audio file where a message is recorded or from where a message is played. Audio files must be .wav files in 44.1 kHz, 16-bit mono PCM format. If you give an invalid filename, it is placed in or taken from the /cxc directory.
- [*greeting*]—A greeting file that may be applied first as a prompt.
- [*cluster*]—When **enabled**, the file is available to all ASCs in the cluster. When **disabled** (the default), the file is only available on the local ASC.

call-control-memo-end

Ends a recording on the specified call leg.

Syntax

```
call-control-memo-begin <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-message

Connects to a given endpoint, plays the file you specify, then disconnects the call. If you specify a From URI, that appears in the From header as the calling party; if no URI is specified, the To URI is used as the From header.

Syntax

```
call-control-message <filename> <endpoint> [from] [requestId] [async]
   [config]
```

Arguments

- <*filename*>—The name of the audio file where a message is recorded or from where a message is played. Audio files must be .wav files in 44.1 kHz, 16-bit mono PCM format. If you give an invalid filename, it is placed in or taken from the /cxc directory.
- <*endpoint*>—The URI of the call’s destination.
- [*from*]—The originating SIP URI of the call.

- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

call-control-message-request

Sends a MESSAGE on an existing call.

Syntax

```
call-control-message-request <handle> [content-type] [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-modify

Sends a re-INVITE on an existing call leg.

Syntax

```
call-control-modify <handle> [content-type] [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*content-type*]—Specifies the Content-Type: for the indication.

- [*body*]—Specifies the body for the indication.

call-control-monitor-file

Attaches a monitor session to a recording file. A recording file can be a live session currently being recorded, an old session that was recorded, an on-demand recording of a session, or a memo actively being recorded.

Syntax

```
call-control-monitor-file <handle> <session-id> <monitor-target>
   [seek-offset] [position]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*session-id*>—The optional session ID for the session.
- <*monitor-target*>—The filename of the file to be played. This can be:
    - session—A session recording file is going to be monitored.
    - memo—A memo actively being recorded is going to be monitored.
    - name—The on-demand filename specified in the call-control-record-start action is being monitored.
- [*seek-offset*]—The offset, in milliseconds, to begin seeking. A negative value seeks backwards. Seeking starts at the spot specified by the **position** parameter.
- [*position*]—Indicates the position to begin seeking

call-control-monitor-session

Attaches a monitor session to a live target session. The monitor session must join the target session in-progress as it has no ability to seek forward or backward during a live recording.

Syntax

```
call-control-monitor-session <handle> <session-id>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*session-id*>—The optional session ID for the session.

call-control-mute-off

Turns off the mute functionality for a call leg.

Syntax

```
call-control-mute-off <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-mute-on

Turns on the mute functionality for a call leg.

Syntax

```
call-control-mute-on <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-notify

Causes a SIP NOTIFY message to be sent to the party you specify in the **handle** parameter, with the value of the Event header set by the **event** parameter.

Syntax

```
call-control-notify <handle> <event>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*event*>—The content of the Event header.

call-control-notify-request

Sends a NOTIFY on an existing call.

Syntax

```
call-control-notify-request <handle> <event> [async] [content-type]
   [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*event*>—The content of the event header.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-options-request

Sends an OPTIONS on an existing call.

Syntax

```
call-control-options-request <handle> [content-type] [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-park

Creates a call to an endpoint from a given SIP URI. If you specify a From URI, it is used as the From URI in the SIP message; if you specify no From URI, the From URI is that of the given endpoint.

Syntax

```
call-control-park <endpoint> [from] [requestId] [async] [sessionID]
   [persist] [config]
```

Arguments

- *<endpoint>*—The URI of the call's destination.
- [*from*]—The originating SIP URI of the call.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled** (the default), the OS-E waits for the action to complete before returning a response.
- [*sessionID*]—The optional session ID for a rendezvous session.
- [*persist*]—When **enabled**, a connected session remains parked even when the remote endpoint disconnects.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

Enclose the value in quotation marks when using the CLI.

call-control-park-to-session

Parks a call to an existing session.

Syntax

```
call-control-park-to-session <endpoint> <sessionID> [from] [requestId]
   [async] [persist] [config]
```

Arguments

- <*endpoint*>—The handle of call leg on the existing session.
- <*session-id*>—The optional session ID for the session.
- [*from*]—The From URL of the parked call.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.
- [*persist*]—When **enabled**, a connected session remains parked even when the remote endpoint disconnects.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

call-control-persistence

Makes a call-leg persist in a parked state even when its peer is terminated.

Syntax

```
call-control-persistence <handle> <persist>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*persist*>—When **enabled**, a connected session remains parked even when the remote endpoint disconnects.

call-control-play

Plays a given audio file to the specified call leg. If two call legs are connected, the file is played to both parties.

If the **session-config > media-scanner-settings** is configured, the ASC waits until the recipient (or an answering machine) has finished speaking before delivering the message. If the media scanner times out waiting for the recipient to finish speaking, the file is not played.

Syntax

```
call-control-play <handle> <filename> [startTime] [async]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*filename*>—The name of the audio file where a message is recorded or from where a message is played. Audio files must be .wav files in 44.1 kHz, 16-bit mono PCM format. If you give an invalid filename, it is placed in or taken from the /cxc directory.
- [*startTime*]—The number of milliseconds the ASC waits before playing the file.
- [*async*]—When **enabled**, causes the OS-E to return a response immediately without waiting for the action to complete. When **disabled**, (the default) the OS-E waits for the action to complete before returning a response.

call-control-record-start

Starts the on-demand recording or a target session to a specific <*filename*> file. This recording can then be monitored via the **call-control-monitor-file** action. You can execute this command one or more times for a given target session, provided you give it a different <*filename*> each time. If a <*filename*> already exists for a given target session, the existing <*filename*> is preserved and the action fails.

Syntax

```
call-control-record-start <session-id> <filename>
```

Arguments

- <*session-id*>—The optional session ID for the session.
- <*filename*>—The name of the recording for this particular target session.

call-control-record-stop

Stops the on-demand recording or a target session to a specific <*filename*>.

Syntax

```
call-control-record-stop <session-id> <filename>
```

Arguments

- <*session-id*>—The optional session ID for the session.
- <*filename*>—The name of the recording for this particular target session.

call-control-redirect

Redirects an initiated call to a new endpoint, prior to the call being answered. This creates a new call leg and cancels the original one.

Syntax

```
call-control-redirect <handle> <endpoint> [config]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*endpoint*>—The URI of the call's destination.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

call-control-reject

Rejects an incoming call from an offering endpoint.

Syntax

```
call-control-reject <handle> [response-code] [responseText]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*response-code*]—The SIP response code to return in response.
- [*responseText*]—Text text to return in the response.

call-control-retrieve

Retrieves the held call leg you specify by call handle. This reconnects the call's media for that call leg and, if present, the other call leg.

Syntax

```
call-control-retrieve <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-ringing

Redirects an initiated call to a new endpoint, prior to the call being answered. This creates a new call leg and cancels the original one.

Syntax

```
call-control-redirect <handle> <endpoint> [config]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*endpoint*>—The URI of the call's destination.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

Enclose the value in quotation marks when using the CLI.

call-control-send-message

Sends a message to the endpoint specified by the To URI. If you specify a From URI, it is used for the From URI. If a From URI is not specified, the From URI is the same as the To URI.

Syntax

```
call-control-send-message <to> <from> [requestId] [content-type]
   [body] [config]
```

Arguments

- <*to*>—The destination SIP URI of the call.
- <*from*>—The originating SIP URI of the call.
- [*requestId*]—A unique identifier provided by an external application. This value can be used to identify the call in subsequent events and actions. If a requestId is specified, there is a corresponding XML element in the event messages generated for the session.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.
- [*config*]—The **session-config** on the OS-E to use to process a call. Use the full path to the **session-config**. For example:

```
vsp\session-config-pool\entry MyConfig
```

call-control-subscribe-request

Sends a SUBSCRIBE on an existing call.

Syntax

```
call-control-subscribe-request <handle> [pkg] [expires] [content-type]
   [body]
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- [*pkg*]—Specifies the package for the SUBSCRIBE.
- [*expires*]—The expiration value for the SUBSCRIBE.
- [*content-type*]—Specifies the Content-Type: for the indication.
- [*body*]—Specifies the body for the indication.

call-control-terminate

Terminates the call leg indicated by the handle you specify. This parameter is only available for calls with a parked status.

Syntax

```
call-control-terminate <handle>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.

call-control-transfer

Transfers the specified call leg to the specified To SIP URI. The original call leg, referred to by its handle, is disconnected. Handle can be thought of as belonging to the party doing the transfer, even though the transfer is done via a third-party action.

Syntax

```
call-control-transfer <handle> <to>
```

Arguments

- <*handle*>—Identifies the leg of a session. Handles are returned as part of the <info> element of **call-control** results and can be used to manipulate each leg of a call independently.
- <*to*>—The destination SIP URI of the session.

Boot Media Creator Updates

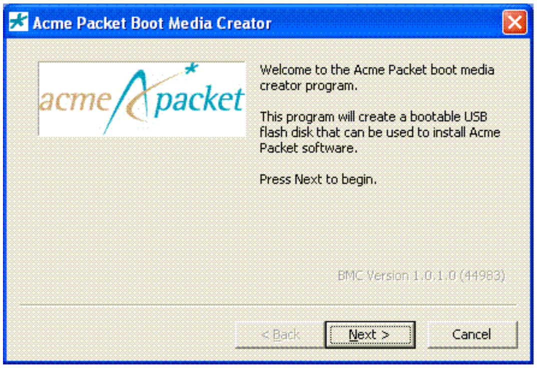
The Boot Media Creator (BMC) is the program that creates the OS-E software installation USB stick. In releases prior to 3.6.0m5p1, the software image files were embedded in the BMC tool. However, the image files have become too large. You must now download both the BMC tool and the USB image file separately.

Before using the bmc.exe program, you must first download both the USB image file you are using to commission your system and the ZIP file containing the BMC tool. Then extract the bmc.exe program from the ZIP file.

To extract the BMC:

1. Click on the file named **nnSE####.bmc.exe**.

2. Click **Run**.
3. Click **Next** to start the BMC.



Acme Packet Boot Media Creator

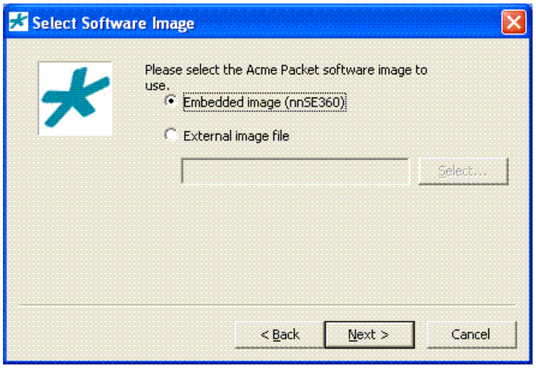
Welcome to the Acme Packet boot media creator program.

This program will create a bootable USB flash disk that can be used to install Acme Packet software.

Press Next to begin.

BMC Version 1.0.1.0 (44983)

4. Click **Next** to display the Select Software Image page.



Select Software Image

Please select the Acme Packet software image to use.

● Embedded image (nnSE360)

○ External image file

5. Select the **External image file** radio button.
6. Click the **Select...** button to browse to the USB image you downloaded and saved. The USB image name is in the format nnSE###-usb.img.gz.
7. Click **Next** to validate the software image.

Configuration Changes in Release 3.7.0

The section provides a summary of the additions, changes, and deletions to the OS-E configuration when upgrading to Release 3.7.0. It covers new objects and properties, configuration objects and properties that have been renamed, and those objects that have been deleted and are no longer available.

New Objects in Release 3.7.0

Object name	Associated properties	Description
grant-pattern		Configures the pattern to use to extract a privilege to grant.
	name	Enter a descriptive name to give this grant. There is no default setting. **Example: set name DeskPhoneEvent**
	pattern	Enter the regular expression pattern to use to define the attribute. There is no default setting. **Example: set pattern "\+1 \((\d{3})\) (\d{3})-(\d{4})"**
	resource-identity	Select the type of matching to use to identify a resource-type. The default setting is **equals**. • equals <*value*>—The value that a user provides during an authorization request must be exactly the same as the resulting resource-identity. This is the default setting. • matches <*expression*>—The value that a user provides during an authorization request is matched against the resource-identity using a regular expression match. • any—Any value a user provides during an authorization request matches. **Example: set resource-identity any**
	regex-type	*Advanced property*. Specify the type of regular expression. • custom—Custom Regular Expressions and Replacements • PCRE—Perl Compatible Regular Expressions and Replacements The default setting is PCRE. **Example: set regex-type custom**

Object name	Associated properties	Description
	resource-type	Select the resource type that this extracted value represents. • call • call-recording • call-monitors • call-media-insertion • event-channel • registration • sip-request • file There is no default setting. **Example: set resource-type call-recording**
	privileges	Select the CRUD privileges to allow for this resource-type. • create • retrieve • update • delete The default setting all. **Example: set privileges update**
default-grant		Configures default grants, which apply to all OS-E users matching the specified resource identity.
	name	Enter a descriptive name to give this grant. There is no default setting. **Example: set name PhoneEvent**

Object name	Associated properties	Description
	resource-identity	Select the type of matching to use to identify a resource-type. The default setting is **equals**. • equals <*value*>—The value that a user provides during an authorization request must be exactly the same as the resulting resource-identity. This is the default setting. • matches <*expression*>—The value that a user provides during an authorization request is matched against the resource-identity using a regular expression match. • any—Any value a user provides during an authorization request matches. **Example: set resource-identity any**
	resource-type	Select the resource type that this extracted value represents. • call • call-recording • call-monitors • call-media-insertion • event-channel • registration • sip-request • file There is no default setting. **Example: set resource-type call-recording**
	privileges	Select the CRUD privileges to allow for this resource-type. • create • retrieve • update • delete The default setting all. **Example: set privileges update**

Object name	Associated properties	Description
group-grant		Configures default and attribute grants for specific groups. Group grants apply to users belonging to these groups and matching the resource-identity.
	name	Enter the name of the group for which you are configuring this grant. There is no default setting. **Example: set name engineering**
	default-grant	Configures a default grant for this group.
	attribute-grant	Configures an attribute grant for this group.
	application-accessible	*Advanced property*. Indicate whether or not to expose this group value externally. true | false The default setting is **true**. **Example: set application-accessible false**
in-ice-settings		Enables and configures ICE settings on the in-leg.
	admin	Enables or disables ICE on the in-leg. enabled | disabled The default setting is **disabled**. **Example: set admin enabled**
	connectivity-check-time-out	Specifies the time in ms before a STUN connectivity check times out. Min: 0 / Max: 4294967296 The default setting is **100**. **Example: set connectivity-check-time-out 150**

Object name	Associated properties	Description
	connectivity-check-max-retransmits	Specifies the number of times the OS-E retransmits ICE STUN connectivity checks before labeling a candidate pair as Failed. To achieve maximum interoperability with Chrome, set this value to no less than 200. Min: 0 / Max: 255 The default setting is **7**. **Example: set connectivity-check-time-out 200**
	delay-stun-responses	*Advanced setting.* When **enabled**, the OS-E does not respond to STUN until the 200 OK is received. enabled | disabled The default setting is **disabled**. **Example: set delay-stun-responses enabled**
	suppress-re-invites	*Advanced setting.*When **enabled** the OS-E does not send a re-INVITE when ICE completes successfully. enabled | disabled The default setting is **disabled**. **Example: set suppress-re-invites enabled**
out-ice-settings		Enables and configures ICE settings on the out-leg.
	admin	Enables or disables ICE on the out-leg. enabled | disabled The default setting is **disabled**. **Example: set admin enabled**

Object name	Associated properties	Description
	connectivity-check-time-out	Specifies the time in ms before a STUN connectivity check times out. Min: 0 / Max: 4294967296 The default setting is **100**. **Example: set connectivity-check-time-out 150**
	connectivity-check-max-retransmits	Specifies the number of times the OS-E retransmits ICE STUN connectivity checks before labeling a candidate pair as Failed. To achieve maximum interoperability with Chrome, set this value to no less than 200. Min: 0 / Max: 255 The default setting is **7**. **Example: set connectivity-check-time-out 200**
	delay-stun-responses	*Advanced setting.* When **enabled**, the OS-E does not respond to STUN until the 200 OK is received. enabled | disabled The default setting is **disabled**. **Example: set delay-stun-responses enabled**
	suppress-re-invites	*Advanced setting.*When **enabled** the OS-E does not send a re-INVITE when ICE completes successfully. enabled | disabled The default setting is **disabled**. **Example: set suppress-re-invites enabled**
in-sdp-attribute-settings		Configures SDP attribute settings on the in-leg.

Object name	Associated properties	Description
	rtcp-mux	Enables or disables RTP/RTCP multiplexing on the in-leg. enabled | disabled The default setting is **disabled**. **Example: set rtcp-mux enabled**
	ssrc-in-sdp	Set to strip to strip out any SSRC information from the SDP. pass | strip The default setting is **pass**. **Example: set ssrc-in-sdp strip**

	patch-audio-group	*Advanced property*. Set to enabled. When the OS-E receives an offer SDP with both audio and video and the line a=group BUNDLE audio video and a response with only audio, it must perform certain functions in order for the audio to work. When enabled, the OS-E performs the following modifications:
		• The OS-E performs RTP/RTCP multiplexing on the in-leg, regardless of the user configuration
		• The OS-E adds bundling information by adding the following to the SDP
		`a=group BUNDLE audio`
		`a=mid:audio`
		• The OS-E generates WebRTC-style SSRC values and adds them to the SDP as well as the RTP/RTCP stream.
		enabled | disabled
		The default setting is **disabled**.
		Example: set patch-audio-group enabled

out-sdp-attribute-settings		Configures SDP attribute settings on the out-leg.

Object name	Associated properties	Description
	rtcp-mux	Enables or disables RTP/RTCP multiplexing on the out-leg. enabled | disabled The default setting is **disabled**. **Example: set rtcp-mux enabled**
	ssrc-in-sdp	Set to strip to strip out any SSRC information from the SDP. pass | strip The default setting is **pass**. **Example: set ssrc-in-sdp strip**

	patch-audio-group	*Advanced property*. Set to enabled. When the OS-E receives an offer SDP with both audio and video and the line a=group BUNDLE audio video and a response with only audio, it must perform certain functions in order for the audio to work. When enabled, the OS-E performs the following modifications:
		• The OS-E performs RTP/RTCP multiplexing on the in-leg, regardless of the user configuration
		• The OS-E adds bundling information by adding the following to the SDP
		`a=group BUNDLE audio`
		`a=mid:audio`
		• The OS-E generates WebRTC-style SSRC values and adds them to the SDP as well as the RTP/RTCP stream.
		enabled | disabled
		The default setting is **disabled**.
		Example: set patch-audio-group enabled

cross-origin-settings		Configures the eventpush-service CORS settings.

Object name	Associated properties	Description
	enable-cross-origin	*Advanced property*. Enables or disables cross-origin requests to the web-server. • **accept-any-origins** • **accept-specified-origins** • **deny-all-origins** The default setting is **accept-any-origins**. **Example: set enable-cross-origin deny-all-origins**
	allowed-cross-origin	*Advanced property*. Specifies the origins whose requests the OS-E accepts when the **enable-cross-origin** property is set to **accept-specified-origins**. There is no default setting. **Example: set allowed-cross-origin http://mysite.com**
cross-origin-resource-sharin g		Configures the web-services' CORS settings.
	enable-cross-origin	*Advanced property*. Enables or disables cross-origin requests to the web-server. • **accept-any-origins** • **accept-specified-origins** • **deny-all-origins** The default setting is **accept-any-origins**. **Example: set enable-cross-origin deny-all-origins**
	allowed-cross-origin	*Advanced property*. Specifies the origins whose requests the OS-E accepts when the **enable-cross-origin** property is set to **accept-specified-origins**. There is no default setting. **Example: set allowed-cross-origin http://mysite.com**
ws-port		Configures the WebSocket listener port.

Object name	Associated properties	Description
	port	Specifies the port number for the listener socket. Min: 0 / Max: 65535 There is no default setting. **Example: set port 5000**
	admin	Enables or disables this listener port. enabled | disabled The default setting is **enabled**. **Example: set admin enabled**
	resource-path	Specify the HTTP resource to expect in the HTTP GET message. The default setting is **/sip**. **Example: set resource-path /sip**
	http-authentication	Specifies the type of authentication mode to use for the HTTP handshake. Valid values are: • none—No HTTP authentication for WebSockets • local—Locall HTTP authentication for WebSockets • radius—RADIUS HTTP authentication for WebSockets • diameter—Diameter HTTP authentication for WebSockets • directory—Directory HTTP authentication for WebSockets (this references the vsp > enterprise > directories configuration) • accept—Accept all HTTP authentication for WebSockets • reject—Reject all HTTP authentication for WebSockets The default setting is **none**. **Example: set http-authentication local**

Object name	Associated properties	Description
	http-authentication-realm	When enabled, the realm to use for HTTP authentication. There is no default setting. **Example: set http-authentication-realm ws1**
wss-port		Configures the WebSocket Secure listener port.
	port	Specifies the port number for the listener socket. Min: 0 / Max: 65535 There is no default setting. **Example: set port 6000**
	admin	Enables or disables this listener port. enabled | disabled The default setting is **enabled**. **Example: set admin enabled**
	resource-path	Specify the HTTP resource to expect in the HTTP GET message. The default setting is **/sip**. **Example: set resource-path /sip**

Object name	Associated properties	Description
	http-authentication	Specifies the type of authentication mode to use for the HTTP handshake. Valid values are: • none—No HTTP authentication for WebSockets • local—Locall HTTP authentication for WebSockets • radius—RADIUS HTTP authentication for WebSockets • diameter—Diameter HTTP authentication for WebSockets • directory—Directory HTTP authentication for WebSockets (this references the vsp > enterprise > directories configuration) • accept—Accept all HTTP authentication for WebSockets • reject—Reject all HTTP authentication for WebSockets The default setting is **none**. **Example: set http-authentication local**
	certificate	Specifies a certificate to use for this listener port. This property references a certificate configured under the **vsp > tls > certificate** object. If you do not specify a certificate, the OS-E uses the SIP certificate. There is no default setting. **Example: set certificate vsp\tls\certificate cert1**
subscribe-service-settings		Configures settings for the subscription-based notification service. This service caches subscription information and allows external applications to more easily send and receive NOTIFY Events.

Object name	Associated properties	Description
	cache-bindings	Specify if the subscription-based notification service should cache subscriptions or not. enabled | disabled The default setting is **disabled**. **Example: set cache-bindings enabled**
	accept-event-locally	Specify the subscription events that are handled locally. There is no default setting. **Example: set accept-event-locally PhoneEvents**

New properties in Release 3.7.0

Object name	Associated properties	Description
third-party-call-control	ignore-route-headers	*Secondary property.* Specifies whether the OS-E ignores the route header (when present) and forward the request using the request-URI. enabled | disabled The default setting is **disabled**. **Example: set third-party-call-control enabled**
multimedia-stream-settings	jitter-buffer-size	Sets the transmission delay for the multimedia server jitter buffer. Min: 0 / Max: 4294967296 The default setting is **40**. **Example: set jitter-buffer-size 55**
	auto-accept	When enabled, the MSS automatically accepts all calls. enabled | disabled The default setting is **disabled**. **Example: set auto-accept enabled**
permissions	action-filter-whitelist	Select an existing action-filter you want users with this permissions set to have access to. There is no default setting. **Example: set action-filter-whitelist filter1**
user	attribute	Specify attributes that this user has assigned. There is no default setting. **Example: set attribute PhoneNumber**
	group	Specify the groups that this user is a member of. There is no default setting. **Example: set group eng**

Object name	**Associated properties**	**Description**
multimedia-streaming-config > server	delegate-server	Specifies the server configured under the **vsp > enterprise > servers > sip-gateway** object. There is no default setting. **Example: set delegate-server vsp/ enterprise/servers/sip-gateway server1**
event-settings	include-media-content	Enables or disables the inclusion of SDPs in events. enabled | disabled The default setting is disabled. **Example: set include-media-content enabled**
in-encryption	reuse-key	When enabled, the OS-E reuses the initial SRTP key for subsequent SDP exchanges. enabled | disabled The default setting is **disabled**. **Example: set reuse-key enabled**
out-encryption	reuse-key	When enabled, the OS-E reuses the initial SRTP key for subsequent SDP exchanges. enabled | disabled The default setting is **disabled**. **Example: set reuse-key enabled**

Object name	**Associated properties**	**Description**
session-config > registration	broadsoft-registration-survivability-sca	*Advanced property.* When enabled, the OS-E checks the state of the Broadsoft server on the registration plan. If the server is down, the OS-E accepts the SCA registration locally for the duration of the client expiration. For this period, the registration is throttled. If the registration is accepted in this mode, the SCA survivability mapping is created by extracting the phone number from the contact address using the configured regular expression. enabled | disabled The default setting is **disabled**. **Example: set broadsoft-registration-survivability-sca enabled**
data-locations	rtp-on-demand-mixed	Sets the location to which the OS-E writes for playback of on-demand recorded calls. This is where on-demand RTP files are mixed to create files that can then be played back. There is no default setting. **Example: set rtp-on-demand-mixed / recordings**
external-services	max-http-client-connections	Specifies the maximum number of outbound HTTP connects allowed per host. Min: 5 / Max: 100 The default setting is **10**. **Example: set max-http-client-connections 25**
	max-http-connections	Specifies the maximum number of outbound HTTP connections allowed. Min: 100 / Max: 300 The default setting is **100**. **Example: set max-http-connections 150**

Object name	Associated properties	Description
eventpush-service	accept-connections	*Advanced property.* Specify the number of connection requests that can be queued up before the OS-E sends rejections. Min: 0 / Max: 200000 The default setting is **100**. **Example: set accept-connections 55**
	authentication	Select the mode to use for HTTP authentication. • none—do not perform authentication • handshake-extension—Use authentication data passed in the comet handshake extension. The default setting is **none**. **Example: set authentication handshake-extension**
	cross-origin-settings	Configures the eventpush-service CORS settings.
	idle-connection-timeout	*Advanced property.* Specifies the maximum time a connection can idle before it times out and the OS-E drops it. Min: 0 / Max: 200000 The default setting is **20**. **Example: set idle-connection-timeout 25**
	max-sessions	*Advanced property.* Specifies the maximum number of concurrent sessions allowed. Min: 0 (there is no limit) / Max: 200000 The default setting is **10000**. **Example: set max-sessions 10500**
	session-idle-timeout	Specifies the maximum time for a session to idle before it times out and the OS-E drops it. Min: 0 / Max:4294967296 The default setting is **60**. **Example: set session-idle-timeout 45**

Object name	Associated properties	Description
web-service	accept-connections	*Advanced property.* Specify the number of connection requests that can be queued up before the OS-E sends rejections. Min: 0 / Max: 200000 The default setting is **100**. **Example: set accept-connections 55**
	ciphers	*Specify SSL cipher suite names separated by commas.* *There is no default setting.* **Example: set ciphers SSL_RSA_WITH_RC4_128_SHA, TLS_RSA_WITH_AES_128_CBC_SHA**
	cross-origin-resource-sharing	Configures the web-services' CORS settings.
	idle-connection-timeout	*Advanced property.* Specifies the maximum time a connection can idle before it times out and the OS-E drops it. Min: 0 / Max: 200000 The default setting is **20**. **Example: set idle-connection-timeout 25**
	max-connections	*Advanced property.* Specifies the maximum number of connections that the server can accept and process at any given time. When this threshold is reached, the server does not accept any more connections until the number falls below this value. **Note:** Based on the **accept-connections** value, the OS-E may still accept connections into the queue. Min: 0 / Max: 200000 The default setting is **10000**. **Example: set max-sessions 10500**

Object name	Associated properties	Description
	max-keep-alive-requests	*Advanced property.* Specifies the maximum number of HTTP requests which can queued before the connection is closed by the server. Min: -1 / Max:200000 The default setting is **-1** (unlimited). **Example: set max-keep-alive-requests 15**
	max-sessions	*Advanced property.* Specifies the maximum number of concurrent sessions allowed. Min: 0 (there is no limit) / Max: 200000 The default setting is **10000**. **Example: set max-sessions 10500**
	session-idle-timeout	Specifies the maximum time in seconds for a session to idle before it times out and the OS-E drops it. Min: 0 / Max:4294967296 The default setting is **30**. **Example: set session-idle-timeout 45**
web	accept-connections	*Advanced property.* Specify the number of connection requests that can be queued up before the OS-E sends rejections. Min: 0 / Max: 200000 The default setting is **100**. **Example: set accept-connections 55**
	idle-connection-timeout	*Advanced property.* Specifies the maximum time a connection can idle before it times out and the OS-E drops it. Min: 0 / Max: 200000 The default setting is **20**. **Example: set idle-connection-timeout 25**

Object name	Associated properties	Description
	max-connections	*Advanced property.* Specifies the maximum number of connections that the server can accept and process at any given time. When this threshold is reached, the server does not accept any more connections until the number falls below this value. **Note:** Based on the **accept-connections** value, the OS-E may still accept connections into the queue. Min: 0 / Max: 100000 The default setting is **10000**. **Example: set max-sessions 10500**
	max-keep-alive-requests	*Advanced property.* Specifies the maximum number of HTTP requests which can queued before the connection is closed by the server. Min: -1 / Max:200000 The default setting is **-1** (unlimited). **Example: set max-keep-alive-requests 15**
	max-sessions	*Advanced property.* Specifies the maximum number of concurrent sessions allowed. Min: 0 / Max: 200000 The default setting is **30**. **Example: set max-sessions 10500**

Object name	**Associated properties**	**Description**
sip	http-authentication	Specifies the type of authentication mode to use for the HTTP handshake. Valid values are: • none—No HTTP authentication for WebSockets • local—Locall HTTP authentication for WebSockets • radius—RADIUS HTTP authentication for WebSockets • diameter—Diameter HTTP authentication for WebSockets • directory—Directory HTTP authentication for WebSockets (this references the vsp > enterprise > directories configuration) • accept—Accept all HTTP authentication for WebSockets • reject—Reject all HTTP authentication for WebSockets The default setting is **none**. **Example: set http-authentication local**
	http-authentication-realm	When enabled, the realm to use for HTTP authentication. There is no default setting. **Example: set http-authentication-realm ws1**
	ws-port	Configures the WebSocket listener port.
	wss-port	Configures the WebSocket Secure listener port.
gui-preferences	display-management-links	*Advanced property.* When enabled, the OS-E displays management information links on the **Tools** page when **channel** is set to something other than **none**. enabled | disabled The default setting is **enabled**. **Example: set display-management-links disabled**

Object name	Associated properties	Description
directories	to-policy	Specifies the policy to apply to traffic going to this directory's users.
static-stack-settings	domain-alias	Adds one or more aliases for the domain in which the OS-E resides. There is no default setting. **Example: set domain-alias www.abc.com**
route	routing-tag	Labels this arbiter profile for matching against routing tags. There is no default setting. **Example: set routing-tag tag1**
stun-server	transport	Specifies which transport protocol over which STUN messages are exchanged between a SIP endpoint and the OS-E STUN server. Valid values are: • UDP • TCP • TLS The default setting is **UDP**. **Example: set transport TCP**

Renamed Properties in Release 3.7.0

Old name	New name
permissions > action-filter	**permissions > action-filter-blacklist**

MIB Changes in Release 3.7.0

This section covers changes that have been applied to Management Information Base (MIB) object definitions.

New MIB Tables in Release 3.7.0

MIB table name	Description
authorizedUserAttributes	List authorized users'attributes as returned by the directory.
authorizedUserGroups	List authorized users' groups as returned by the directory.
authorizedUserPrivileges	Lists authorized users' authorization privileges that were derived from the attributes and groups returned by the directory.
authorizedUserSummary	Lists a summary of authorized users, the information returned by the directory, and their privileges.
iceStateStatus	Status of ICE state machines.
locationSubscriptionCache	Location subscription cache statistics.
wsListener	WebSocket listener socket statistics.
wssListener	Secure WebSocket listener socket statistics.
webExtStatus	Web ext status.
webServicesPorts	The TCP ports where the web services HTTP server is listening.
webServicesStatus	The web services status.
webServicesVirtualHostDeployableApplications	Detailed information about the virtual host deployable applications.

New MIB Objects in Release 3.7.0

MIB table name	Description
cometdStatus	Cometd service status.
systemDirectories	Base system directories.
webChannelPreferences	GUI channel preferences.

Changed Tables in Release 3.7.0

MIB object name	Description
callingGroupPool	ADDED: callingGroupPoolWebSocketPath, callingGroupPoolWebSocketSecretTag, callingGroupPoolWebSocketUserName
directoryStatus	ADDED: directoryStatusAdminState, directoryStatusOperState

MIB object name	Description
mediaSessionRecord	ADDED: mediaSessionRecordIceConfigIn, mediaSessionRecordIceConfigOut, mediaSessionRecordSdpAttrIn, mediaSessionRecordSdpAttrOut
multimediaStreamingPool	ADDED: multimediaStreamingPoolWebSocketPath, multimediaStreamingPoolWebSocketSecretTag, multimediaStreamingPoolWebSocketUserName
registrationPlan	ADDED: registrationPlanMaxAuthenticationAttempts
registrationRouting	ADDED: registrationRoutingMaxAuthenticationAttempts
routeServerBox	ADDED: routeServerBoxEntries
routeServerControlledActionStatus	ADDED: routeServerControlledActionStatusEntries
sipServerPool	ADDED: sipServerPoolWebSocketPath, sipServerPoolWebSocketSecretTag, sipServerPoolWebSocketUserName
subscriptionListEntry	ADDED: subscriptionListEntryAddrScheme
switchPool	ADDED: switchPoolWebSocketPath, switchPoolWebSocketSecretTag, switchPoolWebSocketUserName
tlsListener	ADDED: tlsListenerIncomingForbidden
vxHostOptions	ADDED: vxHostOptionsBound, vxHostOptionsBoundToMacvlan
vxHosts	ADDED: vxHostsEthernet, vxHostsLocallyBound

Changed Objects in Release 3.7.0

MIB object name	Description
locationService	ADDED: locationServiceDelegatePollInterval, locationServiceMaxDelegateRegistrations
locationSummary	locationSummaryTotalRemoteRegisteredBindings, locationSummaryTotalStaticBindings, locationSummaryTotalStaticRegisteredBindings
routeServerActionStatus	routeServerActionStatusRoutes

New Traps in Release 3.7.0

Trap name
There are no new traps in Release 3.7.0.

Changed Traps in Release 3.7.0

MIB table name	Description
There are no changed traps in Release 3.7.0.	

Known Problems, Restrictions, and Operational Considerations in 3.7.0

Several sip_send_xxxx actions - do not report 'authorization failure' as the result code when authorization actually fails.

The following actions are affected:

- sip-send-message
- sip-send-notify
- sip-send-options
- sip-send-other
- sip-send-subscribe
- sip-send-unsubscribe

The desired authorization functionality (as described in our docs) does not work propertly. Any authenticated (i.e. properly logged in) user can execute these actions. This is generally the case for several hundred actions that we have on the system. In 3.7.0 we've added an authorization service which has the ability to control which users can perform certain actions on certain endpoints. For this release, the authorization functionality will not control these listed actions.

Unable to install 3.7.0 from a USB stick on a Cisco UCS C200

In the mean time, a workaround is to stick the system using 360m5 or previous revision and then perform a supertar upgrade to 3.7.0

If in- and out-dtmf preferences are set to rfc2833, then patch-audio-group generates an SSRC but the kernel does not rewrite the RTP stream.

Patch-audio-group is needed for a specific WebRTC-to-SIP interop scenario. If WebRTC offers audio and video, but SIP answers with only audio, then we need to include the group:BUNDLE and ssrc attributes in the SDP answer sent back to WebRTC. This implies we mustrewrite the SSRC in the RTP stream going back to WebRTC with the SSRC placed in the answer.Path-audio-group installs kernel-rules to do this rewrite.

However, the SSRC rewrite does not happen if dtmf-preferences are configured along with patch-audio group due a conflict with the kernel-rules.

REST version of WebRTC phone does not render media when it originates calls

In Chrome 27 (current stable channel), when the REST WebRTC phone initiates a call, it doesnot render the media received from the remote endpoint. This problem does not occur when using Chrome 26. This happens because REST phone javascript callbacks which execute in Chrome 26 do not execute on Chrome 27. Other WebRTC applications which make proper use the javascript callbacks in Chrome 27 (like the JSSIP sample) are not affected.

call-control-disconnect on an ASC based WebRTC phone to a JsSIP based phone fails to send a BYE to the JsSIP side.

To reproduce, call from the REST WebRTC phone to another endpoint (JSSip phone or regular SIP). Then, hang up on the REST side. A BYE reaches the SIP endpoint, but the session lingers.So in summary, if the REST phone is on the in-leg and the call is terminated from the in-leg, then both endpoints hang up but the session on the ASC lingers.You can cleanup the session by executing the **disconnect-call** action.

WebRTC: SIP to REST call does not work. When calling SIP to REST, if the SIP endpoint is not using encryption then the call fails with a 488. When the REST endpoint answers, it appears the wrong SDP is injected into sdp_action triggering the error.

You can work around this problem by adding the following access configuration for the users where the REST phone logs in as user "restphone"

```
config access
config permissions restphone
set action-filter-blacklist access\permission-filters\action-filter
   noringing  return  config users
config user restphone
set permissions access\permissions restphone
return
return
config permission-filters
config action-filter noringing
set filter call-control-ringing
return
return
return
```

This prevents the **call-control-ringing** action from executing, which avoids the bug.

Diffie-Hellman Logjam Attack Defense

The Diffie-Hellman Logjam attack allows a man-in-the-middle attacker to downgrade vulnerable TLS connections to 512-bit export-grade cryptography, allowing the attacker to read and modify any data passed over the connection.

This attack is similar to the FREAK attack, but is due to a flaw in the TLS protocol rather than an implementation vulnerability. It attacks a Diffie-Hellman key exchange rather than an RSA key exchange. When using the OS-E Web Management System, the attack affects any server that supports DHE_EXPORT ciphers and affects all modern web browsers.

> **Note:** For more information on the Diffie-Hellman Logjam attack, see *https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2015-4000*.

To protect against the Logjam vulnerability:

1. Select the **box** > **interface** > **ip** > **web** object.
2. Set the **ciphers** property to the following:

```
TLS_RSA_WITH_AES_128_CVC_SHA, TLS_RSA_WITH_AES_256_CVC_SHA,
   TLS_RSA_WITH_3DES_EDE_CVC_SHA, TLS_RSA_WITH_RC4_128_SHA
```

> **Note:** If the OS-E is part of a cluster configuration, set the **cipher** property in the **cluster** > **vrrp** > **vinterface** > **ip** > **web** object as well.

3. Click **Set**. Update and save the configuration. No restart is necessary and the changes take effect shortly.

Problems, Restrictions, and Considerations from Prior Releases

Upgrading to Release 3.7.0

- Currently, cluster and controlled upgrades from any release prior to Release 3.6.0 are not supported. Perform the upgrade procedure on each individual box in the cluster. (ID 15131)
- If you are currently running a release prior to 3.3.8, 3.4.2, or 3.5.1, you will need to perform the upgrade to Release 3.7.0 from a USB stick. Refer to the Oracle Communications OS-E Installation and Commissioning Guide for information on creating the USB stick and commissioning the OS-E device.
- If you are currently running Release 3.3.8, 3.4.2, 3.5.1, or later you can perform the upgrade to Release 3.7.0 using the procedure covered in the section, "RADIUS Attributes, CDRs, and RADIUS Servers," using the Release 3.7.0 tar file, or you can perform the upgrade from a USB stick. This means that you can choose either procedure, however, Oracle recommends that you apply the upgrade from the USB stick for better compatibility with future upgrades.
- If the OS-E device had data drives mounted on the original version of software, these data drives will no longer be mounted after the upgrade to Release 3.7.0. Run the **add-device** action to restore the data drives to operate with the new software by specifying the *data-1* or *data-2* drive position and the relevant file system.
- When upgrading a OS-E from a USB stick, the configuration, license, certificates and other components are preserved. However, any data on the RAID-10 data-1 drive is not preserved during this operation as the RAID array is always re-configured, with its data erased. If you require the contents of the RAID-10 data-1 drive, perform the upgrade to 3.7.0 using the procedure, "RADIUS Attributes, CDRs, and RADIUS Servers."

  This does not affect any other platform, including 3rd-party platforms with RAID configured. (ID 14736)

USB Stick Restrictions

If you are upgrading an existing OS-E device from a USB stick, check the /cxc directory for .cfg and .xml files that are larger than 2 MB. Files that are larger than 2 MB will not be backed up to the USB stick and restored during the upgrade process.

All *.cfg and *.xml files in the current working directory (/cxc) less than 2 MB in size are backed up to the stick and restored during the upgrade. (ID 13207)

It is important to remember to remove the USB stick once an upgrade is completed in order to maintain the correct modified configuration. (ID 15640)

Virtual Interfaces per Physical Ethernet

Each physical Ethernet interface supports up to 14 virtual (VX) interfaces.

IP Interfaces per Physical OS-E Device

Release 3.7.0 supports a maximum of 4096 named IP interfaces per OS-E device.

CDR Values on External Databases

When sending accounting CDRs to external databases, values that are unsigned 32-bit integers are stored as signed 32 bit integers in the database record. If the value of the field is larger than 2147483647 and retrieved as an integer, the value is stored as a negative number.

To decode the negative number, add 2^{32} or 4294967296 to the value.

The following columns are affected:

- Duration
- PacketsReceivedOnSrcLeg
- PacketsLostOnSrcLeg
- PacketsDiscardedOnSrcLeg
- PdvOnSrcLeg
- MaxJitterOnSrcLeg

- LatencyOnSrcLeg
- MaxLatencyOnSrcLeg
- PacketsReceivedOnDestLeg
- PacketsLostOnDestLeg
- PacketsDiscardedOnDestLeg
- PdvOnDestLeg
- MaxJitterOnDestLeg
- LatencyOnDestLeg
- MaxLatencyOnDestLeg
- Rx1000FactorOnDestLeg
- Rx1000FactorOnSrcLeg
- huntingAttempts
- callPDD

(ID 15898)

Modifying the Timezone

When the timezone property is modified in the Box configuration several Java processes must be manually restarted on that OS-E to pickup the modification. The java processes that need to be restarted are:

- Web
- DIR
- WS
- Acct
- Presence
- Eventpush
- DOS

(ID 15640)

Using the Configuration Import Utility

The import and conversion utility that allows you to move the configuration file (*cxc.cfg* by default) to other OS-E devices. This solves problems associated with managing MAC addresses from one system to another anytime the configuration file is transferred. The import utility uses the XML transform program to run the conversion.

Perform the following steps:

1. Save the current configuration to XML format to a USB stick. The new file is named *template.xml*.

```
NNOS-E>> config save xml /mnt/usb/template.xml
```

2. Insert the USB stick into the USB port on the OS-E master system to which the template.xml file is imported.

3. Run the XML transform program to import the **template.xml** file.

```
NNOS-E>> xml transform cfg-import.xsl /mnt/usb/template.xml new.xml
   “box1=11:11:11:11:11:11 box2=22:22:22:22:22:22
   box3=33:33:33:33:33:33 box4=44:44:44:44:44:44”
   Success
```

   where *cfg-import.xsl* is the name of the style sheet included with OS-E, *template.xml* is the name of the original *cxc.cfg* file (saved as XML), and *new.xml* is the resulting name of the new configuration file that you just imported to OS-E. Included in quotation marks (“) is the list of MAC addresses to which the new configuration file is imported.

4. At the master OS-E device receiving the new configuration file, replace the running the configuration file with the *new.xml* file. If the new configuration is operating as expected, execute **config save**. The four devices in cluster will automatically receive the new configuration file.

```
NNOS-E>> config replace new.xml
NNOS-E>> config save
```

Installing the Cisco JTAPI Jar File

This is needed only for customers who are using the external presence JTAPI communication feature on a OS-E device interoperating with the Cisco CallManager.

Perform the following steps to install the Cisco JTAPI software.

1. Log in to the computer where you want to install the Cisco JTAPI client software.
2. Close all Windows programs.
3. Open a Web browser.
4. Go to the Cisco CallManager administration windows at:

   http://*name*/CCMAdmin/main.asp

   where:

   name specifies the name or IP address of the Cisco CallManager.

   Note: If the above web address does not access the Cisco CallManager administration window properly, try using the following:

   http://<call manager>/plugins/jtapi.jar
5. Choose **Application->Install Plugins**.
6. Choose the **Cisco JTAPI** link.
7. Save the file on your desktop and follow the instructions in the pop-up windows.

Note: Install Cisco JTAPI software on the default drive as directed by the installation software. When Windows NT is installed in C:\\SINNT, the default directory, for example, is C:\\WINNT\Java\lib.

At the platform or blade running OS-E, perform the following steps:

8. Copy the *jtapi.jar* from the **Windows\Java\lib** directory to **/cxc_common/jtapi** and rename the files to the following:
   - Cisco 4 CallManager **jtapi jar** to **jtapi-cisco-2.1.jar**
   - Cisco 5 Call Manager **jtapi.jar** to **jtapi-cisco-3.0.jar**
   - Cisco 6 Call Manager **jtapi.jar** to **jtapi-cisco-4.0.jar**
9. Restart the presence process by performing a **restart warm**.

Routing to Location Cache When Destination Server is “Down”

In Release 3.7.0, calls are no longer routed to the location-cache if the destination server is detected as “down” during failover-detection. If there is a matching dial-plan, OS-E will now return a SIP 503 (Service Unavailable) message rather than route the call through the location-cache and returning a SIP 404 message.

Previously, when all the servers on a route were down, the route was removed from the active routing table, causing failure of the dial-plan match and returning a SIP 404.

Virus Scanning

All virus scanning functionality (to include McAfee and icap-server) has been removed and is no longer supported in OS-E.

OS-E Virtual Machine Limitations

- Transcoding is not supported on the VM.
- Feature options that require fine-grained timing such as music-on-hold and announcements may not work properly in the virtual environment. This is due to virtual OS timing issues that are beyond the control of the Oracle software.   If you plan on using these features as part of your application, please contact your Oracle sales representative for further information.

Accounting Reset

When directing accounting records to an external database target, you will need to execute the **accounting reset** action if you edit the database secret password after OS-E has started forwarding records to this database. Otherwise, OS-E will not be able to contact the external database.

The external database password is configured under the **vsp accounting database group server** object using the **password-tag** property. (ID 15403)

Combination of Ringback-File and Call Introduction

Currently, if a ringback file and a call introduction are configured simultaneously, the call introduction is played immediately, followed by the ringback file. As a result, the call recipient never hears the introduction, and the call originator hears the introduction before the ringback file is played.

When operating correctly, the call introduction is played after the call is connected so that both the caller and the call recipient hear it. (ID 13282)

Web Service Pushlets Over HTTPS

Currently, Web service pushlets and external event service applications with a self-signed certificate will not operate over HTTPS connections. (ID 13421, 14394)

Cisco CallManager Interoperability — Automatic Call Forwarding

When Cisco CallManager (CCM) over H.323 is handling an automatic call forward with **inbound faststart** on the CCM disabled, OS-E sends a CCM non-responding **termCap** when handling remote ringback.

For automatic call forwarding to work properly, ensure that CCM **inbound faststart** is enabled. (ID 13748)

Inleg and Outleg TOS Values

When editing the session configuration **sip-settings\inleg-tos** and **outleg-tos** overwrite value settings, specify a number that represents the 8-bit Differentiated Services (DS) field of the IP packet in decimal format, such as 26 for 00011010, and 104 for 01101000. The default setting for both of these properties is 0. (ID 14110)

Google Gadgets and OS-E Management System Browser Windows

When running the OS-E Management System and iGoogle (with Gadgets) simultaneously, make sure that you are run the OS-E Management System and iGoogle in separate browser windows. (ID 14450, 14451)

Accounting

- Currently, when directing CDRs to a RADIUS target, call field filtering (as configured with the vsp\radius-group\call-field-filter object) does not work. By default, all fields are sent to the RADIUS target. (ID 14893)
- With the vsp\accounting\file-system\path **roll-over** property set to *daily*, the OS-E software currently creates a new accounting file each time the system is restarted, resulting in multiple accounting targets per day. All files are sent to the target, so the target receives all CDRs meant for it. (ID 14568)

- If you disable an accounting target that is referenced by an accounting policy in any session configuration and then re-enable the target, accounting will not send the records that accumulated while the accounting target was disabled. (ID 15044)
- For CDRs to be properly sent to an external MS SQL database server, the **box\hostname** property must be specified using an IP address or fully qualified domain name (FQDN). If a partial hostname is specified, then the domain name must be qualified using the **vsp\static-stack-settings**. (ID 13292)
- Currently, the **accounting purge** action is operating slowly, causing call records to consume large amounts of disk space. Oracle recommends that you set up a second disk for accounting records and set the **accounting-root-directory** (under services\data-locations) to that new drive and directory location. Additionally, set the vsp\accounting **retention-period** to 1 (one day) to ensure frequent purging.

  To enable a second drive, perform the following steps:

  1. Unmount, format, and mount the new target drive, such as *data-1*.

```
NNOS-E>> umount data-1
Success !
NNOS-E>> format data-1 reiser-3
Are you sure (y or n)? y
Success!
NNOS-E>> mount data-1
Device is mounted.

NNOS-E>> show mounts (to display the data-1 drive in the list)
```

  2. Under services\data-locations, set the **accounting-root-directory** property to the new target drive and directory.

```
config data-locations> set accounting-root-directory /cxc_common
    data-1/accounting
```

  3. Set the vsp\accounting **retention-period property** to 1 (one day) to ensure frequent purging of accounting records and free disk space.

  (ID 14905)

Generic JDBC Driver

If you are sending CDRs to an external MySQL database server group, (where type is generic), you will need to download and copy the MySQL driver to the OS-E **/cxc/lib/jdbc** directory to properly connect to the MySQL database.

The MySQL 5.1 download is available from the following link.

http://dev.mysql.com/downloads/mysql/5.1.html

- mysql-connector-java-5.1.7-bin.jar

After copying the driver over to the OS-E device, restart the accounting process by issuing a **restart warm**.

At the MySQL database server (Windows XP, NT, Linux), you will need to open port 3306 to allow client access, as follows:

1. **Start->Control Panel->Windows Firewall**.
2. Select **Exceptions** and **Add Port**
3. Add the port name *mysql_port* and port number *3306*.

Note: Follow this procedure for any JDBC driver used with accounting generic database target type.

Removing and Adding Network Interface (NIC) Cards

Whenever an Ethernet network interface is removed and reinstalled on an OS-E you need to perform either the **install nic** or the **install nic-reinitialize** action to reassign Ethernet port numbering. Not doing so could result in a system deadlock.

- **install-nic** — Adds a NIC card to a system that was previously running. Interfaces on the new card are assigned “next” available Ethernet interface numbers.
- **nic-reinitialize** — Removes all the existing NIC interface assignments and rebuilds the interfaces in order. For hardware, the order is based on how the PCI buses are scanned.

  When upgrading or replacing a NIC card with the same number of ports, either **install-nic** or **nic-reinitialize** action may be used. (ID 14333)

H.323 Call Details in the Call Logs

To view H.323 call details from the OS-E Management System Call Logs, select **Sessions,** then select **View->Other** at the right side of the page. Currently, the **H323 Messages** function does not work. (ID 14852)

H.323 Operational Issues

Currently, unanchored calls over H.323 networks (most deployments) will result in remote ringback, call hold and release, call transfer, and music-on-hold (MOH) failures. (ID 14490, 14519, 14523, 14927, 14977)

CUCM-SIP and ACM-SIP Interoperability — Calls on Hold

With interoperating SIP environments involving Cisco Unified Communications Manager (CUCM) and Avaya Call Manager (ACM), set the session configuration third-party-call-control **reinvite-delayed-offer-wait-on-ack** setting to *enabled* and the session configuration **in-hold-translation** and **out-hold-translation** offer and answer attributes to *sendrecv*. Otherwise, inconsistent re-INVITE behaviors will result when calls are placed on hold in these environments. (ID 15091)

Multiple VLANs on VRRP Networks

If you configure multiple VLANs on a VRRP network, and if you have a VLAN configured on that physical interface, you will need to create corresponding (“phantom”) VRRP VLANs on the Ethernet physical interface to enable traffic to reach the VRRP network. However, if the Ethernet interface does not have a VLAN configured, then there is no need to configure the corresponding “phantom” VRRP VLANs. (ID 12378)

OS-E Management System — Configuration Change Indication

When exiting the OS-E Management System, the software does not currently post a configuration change indication if additions and edits were applied with a **Set** or **OK** selection during the active session. Proceed to save or cancel the configuration, as desired. (ID 3920)

Proxy Re-Registration of SNOM Phones

In a proxy registration configuration involving Broadworks and SNOM phones, the first re-register of the phone allows the unregister/register requests to be handled in the correct order. On a second re-register of the phone, the register requests are handled out of order, forcing the phone into the "in-service" state. A third re-register of the phone returns it to the registered state.

DNS and ENUM

The following notes summarize operational issues with the DNS and ENUM functionality.

- The **vsp/enum/resolver** and **vsp/enum/mapping** objects have been removed from the configuration. Both DNS and ENUM servers are now configured using the **vsp/dns/resolver/server** object using the server IP address and the **type** property (dns-only, enum-only, or both).
- The **vsp/dns/resolver/server** entries no longer have the **sip-location** setting. This setting is now a per-session setting that applies to **routing-last-resort-dns** and is configured in **session-config/dns-client-settings**.
- The new **vsp/dns/enum-mapping** requires a domain-name to be specified, replacing the previous **vsp/enum/mapping** object. An upgrade puts e164.arpa as the domain-name.
- An **enum-domain** can not be referenced per server in **vsp/dns/resolver/server**. Use the **vsp/dial-plan/normalization/condition-list/enum-server** configuration. (ID 12881)

Archiving

In Release 3.7.0, an accounting target of the local database must be configured in order for archiving to work. (ID 12883)

Directory and Master Services

If the directory Service is configured and enabled in a cluster environment, the directory service and master database must be configured on the same system before upgrade installation.

Directory service and master system database must be configured and co-exist in the same box. In 3.7.0, the directory service communicates with the master system database and instructs master system database to load users data from the local file system where the directory service is running. If the Directory service and the master system database do not co-exist on the same device, the master system database will no be able to load files generated by the directory service.

Additionally, directory services database tables are not populated to the backup device. In a cluster, OS-E does not support directory services failing over to the backup box. This is because user ID numbers get regenerated and may not match IDs stored in the database for past traffic. (ID 13203)

Monitor-Groups

Currently, the **vsp/monitor-group** and the **media/monitor** *monitor-group* reference are not currently operational. In the OS-E Management System, this affects the **Call-out** function found under Call Logs/Sessions, User Sessions, and Accounting, and when selecting the **Set up playback** template from Call Logs/User Sessions. (ID 13425)

Siemens Fujitsu RX100 and RX300 Servers

When running OS-E on Siemens Fujitsu RX100 and RX300 servers, the onboard Ethernet ports (two) on these servers not currently supported in Release 3.7.0. (ID 13294)

Media Verification Issue

When using media-verification, if call endpoints do not agree upon a packet interval (ptime), the media-verification may end up dropping RTP packets as outside the range for that CODEC/packet interval. (ID 12381)

Inleg and Outleg TOS Values

When editing the session configuration **sip-settings\inleg-tos** and **outleg-tos** overwrite value settings, specify a number that represents the 8-bit Differentiated Services (DS) field of the IP packet in decimal format, such as 26 for 011010, and 104 for 01101000. The default setting for both of these properties is 0. (ID 14110)

Call Monitoring and Transcoding — No Audio

Currently, there is no audio heard between call endpoints if both transcoding and attendant call monitoring are configured in combination. This problem will be addressed in a later release. (ID 12387)

Policy Manager Running Over WebSphere

When running Policy Manager over WebSphere, note the following:

- An HTTP 500-Internal Server error will occur when retrieving any status using Call Manager and the Status application. This problem only occurs if there are no users and permissions configured under the **access** object. Be sure to configure OS-E user names and passwords for the required Web services authentication as described in the manual, Oracle Communications OS-E Management Tools.
- Ensure that the proper web-service credentials are provided when making web service requests to the OS-E domain. Otherwise, OS-E will return an HTTP 401 (Unauthorized) response, causing the WebSphere Policy Manager application to hang. (ID 11619)

Third Party Call Control (3PCC) Call Transfers

During third party call control (3PCC) sessions involving a Cisco Call Manager server, a call transfer involving multiple recipients will result in a new call control window at first call transfer recipient. Normally, the original call control window should remain active without a refresh. (ID 11298)

SIP Server Pools

Currently, the **show sip-server-pool** command does not display the number of out packets; a 0 count is reported. (ID 11504)

Virtual Machine Uptime Reporting

Currently, if you shut down and then restart the OS-E Virtual Machine on the same device, executing the **show system-info** command will report the new VM with an incorrect uptime. The command shows the uptime starting with the statistic associated with previous VM instead of beginning at zero uptime. (ID 11396)

Attendant Call Monitoring

- With anchoring, recording, and call monitoring enabled, OS-E records the call for the call participants (caller A to caller B), but does not record the call session with the third-party attendant (C). Although the call log shows a separate entry for the INVITE from A to C, the Play field is greyed out, indicating no recording. (ID 9542)
- Currently, an intermittent SIP trace error has been observed with locally-registered phones when the call attendant endpoint picks up while the dialed phone is still ringing. (ID 10989)

QoS Call Duration Statistics

Currently, QoS average call duration and post dial delay statistics for endpoints are not being reported (displaying 0 with the **show switch-pool -v** command). (ID 11549)

H.323 — SIP Directive

The session configuration **sip-directive/directive refuse** setting is not currently applied in H.323 to SIP sessions. (ID 11925)

Unmatched Sessions Returned When Searching by Date/To/From

When searching the OS-E Management System **Call Logs->Sessions** using the Date/To/From criteria, a partial fromURI field is shown in the display, causing the matched sessions to appear “unmatched” in the search results. (ID 11868)

Call Field Filtering on Jitter and Media CDR Fields

To properly display jitter and media CDR fields that are added with the **vsp\accounting\database\group\call-field-filter** object, select PDV on the call-field-filter to display the jitter fields, and select RFACTOR under media fields to display the rfactor fields. (ID 11866)

No Audio Available to Call Monitors

Currently, audio is not being sent to SIP phone third-party endpoints configured in the VSP **monitor-group**. (ID 11951)

Microsoft LCS to IBM Sametime

OS-E users federating IBM Sametime and Microsoft LCS may experience issues with federated presence when moving to 3.4.1 and using their previous configuration. Specifically, **sip-settings** can no longer be used to configure the transport for federated traffic. The **request-uri-specification** should be used instead. (ID 11950)

Admission Control Behavior Changes

To address call admission control behaviors, the following admission control settings are now *disabled* by default:

- **registration-admission-control** — If enabled, the controls set with the pending registration high- and low-watermarks are applicable. This admission control suppresses new registrations to allow resolving registrations in progress, preventing “rate of registration” attacks.
- **call-admission-control** — If enabled, allows call admission control (CAC) on OS-E. The following settings are only applicable if **call-admission-control** is enabled:
  - cac-max-calls
  - cac-max-calls-in-setup
  - cac-min-calls-in-setup
  - cac-max-number-of-tls
  - cac-max-tls-in-setup
  - calls-cpu-limit
  - call-response-code-at-threshold
  - call-response-string-at-threshold

  When disabled, only the **static-stack-settings max-number-of-sessions** property controls setup and connection limits.

The following threshold settings have also been modified:

- **registrations-high-cpu-threshold** — Default is 90%. Sets an upper threshold, as a percentage, for registration processing average CPU usage. The registration dynamic threshold is calculated based on the admission-control/ **pending-registrations-high-watermark** property. When the average CPU usage exceeds this high threshold, OS-E decrements the dynamic threshold by 10% until it reaches the value set with the **pending-registrations-low-watermark** property.
- **registrations-low-cpu-threshold** — Default is 70%. Sets the low-end threshold, as a percentage, for registration processing average CPU usage based on the registration dynamic threshold. When the SIP process CPU falls to the low threshold, OS-E increments the threshold by 16% if the average CPU is less than the low threshold and by 4% if less than the high watermark.

Audio Viewer — Audio Loss During Playback

The Archive Viewer may loose audio or video during playback if the RTP stream switches to a CODEC not supported by the Archive Viewer. (ID 9256)

Location-Cache Changes Not Taking Effect

If you edit the **location-call-admission-control** settings in the session configuration, the changes do not take effect after updating and saving the configuration. Any **location-call-admission-control** changes in the session configuration will require a **location-database flush** action for the new settings to take effect. (ID 10801)

Apply-To-Methods Settings

For REGISTER-based sessions and existing registered endpoints, if you edit the **apply-to-method** setting in the session configuration, the change does not take effect after updating and saving the configuration. Any **apply-to-method** change in the session configuration will require a **location-database flush** action for the new setting to take effect. (ID 11149)

H.323 Protocol

High availability support (call failover) is not supported. (ID 10966)

Identical SSH Host Keys

In order to secure access through SSH, Oracle recommends that you periodically run the new **ssh-regenerate** action to create unique SSH host keys on each platform running OS-E software. The action will initiate a cold restart of the system.

New software installations from USB sticks will automatically generate new and unique SSH host keys at installation time.

After running the **ssh-regenerate** action, some SSH clients may experience a problem when making a secure connection to the system.

Over Putty:

The SSH client will display a pop-up window with the message "WARNING - POTENTIAL SECURITY BREACH!" and explain that the server's host key does not match. Since the host key was changed, the correct action is to click **Yes** to begin the Putty session.

Over Linux-Based SSH:

The SSH client will display the following banner followed by a series of messages:

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@   WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED! @

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

To correct this problem, review the series of messages and edit the file specified in the "Offending key in" line at the line number indicated. You can delete this line and then save the file. When the SSH client is run again, the new host key will be added to the known hosts file. (ID 10909)

Large Configurations with 4K VLANs or More

When running large configurations containing 4000 VLANs or more, OS-E may take several minutes to load at a system **restart,** or from other operations involving the configuration file, such as a **config replace.** Use the **cpu-monitor** action to observe CPU usage during configuration loading or any time while OS-E is running. When the CPU usages falls below 10%, the configuration has successfully loaded. (ID 10050)

Call Recording and File Mirroring Limits

When performing call recording and file mirroring, the system now caps the number of files at 50,000 to prevent an excessive number of stored records and overconsumption of system and memory resources. You should configure periodic maintenance (**services/tasks**) to remove old records at regular and scheduled intervals. Optionally, Oracle provides a software license that will allow you to increase the call recording and file mirroring capacity to 200,000 files. (ID 10036)

TFTP Servers

Configuring a TFTP server on a OS-E interface over UDP/TFTP port 69 may result in TFTP not working. For information on properly configuring a TFTP server, Oracle recommends that you run the WinAgents download from the following Web location.

http://www.winagents.com/en/solutions/tftp-over-firewall.php

(ID 10293)

Calling Group Address Limitation

In vsp/calling-groups/group, the **max-number-of-addresses** property sets the maximum number of AORs that can be associated with a calling group. However, currently only one AOR can be associated with a calling-group.

Licensed Features Display

When displaying features from the OS-E Managment System **Services->Features** page, fields that appear in the greyed-out state showing the default values currently being enforced cannot be edited from this page. This includes the “no royalty” CODECs and other licensed features.

The fields that appear greyed-out require the Oracle license update that includes the field(s) to be configurable. (ID 10496)

SIP Tracing During System Load

The OS-E tracing functions are designed to be used in short duration, isolated troubleshooting conditions. These tools should not be used when the system is under heavy SIP traffic load, and doing so may cause system deadlocks and crashes. This is a tool intended for use only with the assistance of Oracle. (ID 9875)

Outbound Local Port Setting

The session configuration **sip-settings/outboundLocalPort** property is not supported with the new tag routing feature in Release 3.3. (ID 8746)

Multiple Unique Media Streams

With some SIP phones, OS-E (with media anchoring enabled) may return two unique media streams, one disabled, and one accepted. This happens in cases where the SIP phones offer multiple media descriptions for alternate SRTP/clear RTP.
(ID 8899)

Alter-Contact Setting Overriding Sip-Settings/OutboundLocalPort

The **registration-plan/route alter-contact** property setting is currently overriding the session configuration **sip-settings/outbound-local-port** setting used for communicating with the peer. The **alter-contact** setting should only select a port if it is set to trunk-port-per-aor or trunk-port-per-binding. (ID 8749)

Local Enterprise Directory User Files

Local user files for XML and CSV directories, as well as introduction and periodic announcement files, should be stored in a folder under the OS-E-recognized directory named **/cxc_common**. If you place local user files in an unknown or unique directory, OS-E will not be able to locate users after you upgrade the device to the current release. (ID 5578)

SIP Sessions

- The SIP transport is currently missing a time-out setting for TLS and TCP connections. A connection is dropped only if the connection is terminated by the remote SIP client or SIP server.
- During the SIP call session, **disconnect-call** and **terminate-call** actions only operate in back-to-back mode and not in proxy mode.
- The condition-list associated with a policy rule is applied to the first SIP call session only.
- Session termination due to a media verification failure will not work when OS-E is operating in proxy mode. This includes voice and video calls created using Windows Messenger. Calls using Sametime or regular SIP phones are not affected. The **terminate-session** setting in the media verification object (vsp/media-verify-config <name>) contains the configuration setting.

EyeBeam Softphone with Rport Option Turned On

The EyeBeam softphone puts a contact address in the INVITE with a port that is different from where the INVITE is sent. Any requests within the dialog originating from the far end are sent to this contact address. This behavior is correct but the EyeBeam phone is ignoring the request in this case. (ID 8286)

Preventing Call Routing Loops

Forwarding loops can occur when a user agent (UA) registered over OS-E, in delegate or local mode, sends an INVITE to a delegate server, and then that delegate server sends the INVITE back to OS-E for further routing. In some cases, the INVITE from the delegate server might be destined for a UA that is also registered over OS-E. Administrators should ensure that if an INVITE is destined for the UA, that the INVITE matches an existing dial-plan, with the dial-plan **location-match-preferred** property set appropriately. Depending on your network routing, this tells OS-E about the order in which it should attempt a match with the **location-cache** rather than with the **dial-plan**.

A routing loop can also occur if OS-E forwards an INVITE to a delegate server, and that delegate server returns the INVITE to OS-E. In some cases, the inbound INVITE may match the same dial-plan, causing a routing loop where both OS-E and the delegate server continue to send INVITE sessions to each other for the same destination until the **max-forwards** property setting has expired to 0. This causes OS-E to allocate several hundred media ports (where OS-E is anchoring the call), quickly exhausting resources if several UAs are participating in the loop.

OS-E administrators can reduce the possibility of routing loops by performing the following tasks:

- Enable the **vsp/location-service/admission-control** to prevent an address-of-record (AOR) from placing more than **vsp/location-service/max-concurrent-calls-per-AOR** setting for simultaneous calls.
- Enable the **vsp/location-service/emission-control** to prevent an AOR from receiving more than **vsp/location-service/max-concurrent-calls-per-AOR** setting for simultaneous calls.

These settings are only active when the **unregistered-sender-directive** is not set to *allow* for the AOR in question. The **unregistered-sender-directive** property must be set to *refuse* in the **vsp/enterprise/servers/sip-gateway** configuration. Setting the **unregistered-sender-directive** in the **pre-session-config** has no effect.

OS-E returns a “503 Server Unavailable “message to an INVITE if the AOR attempts to place more than the supported number of concurrent calls when these settings are enabled.

Additionally, if OS-E acts as a back-to-back user agent (B2BUA) between several user agents, routing loops may occur. In this configuration, OS-E forwards an INVITE to a server, and then that server sends it back to the OS-E device where OS-E again forwards the INVITE on to another server. In some cases, this may not be a routing loop but a valid routing configuration. OS-E administrators can prevent invalid routing loops by editing the **vsp/enterprise/servers/sip-gateway *name*/ loop-detection** property.

RADIUS Authentication and Server Priorities

If you set the **vsp/radius-group/authentication-mode** to *prioritized*, be sure to change the **vsp/radius-group/server** *priority* setting on any configured RADIUS group servers. All previously configured servers inherit the default value of 1. Without setting different priority values, OS-E randomly selects from these servers and ignores the prioritized mode. The system will generate an event indicating that multiple servers have the same priority.

Media Transcoding

- When performing transcoding, OS-E drops RTCP (regardless of the setting of the media **rtcp** property). OS-E records RTCP according to the session-config/ media/ **rtcp log** setting, but does not forward it since the transcoding may change the synchronization source (SSRC) of RTP along the way. If RTCP were forwarded, it may cause problems for endpoints because the stream described with RTCP may not match the RTP packets sent and received.
- When OS-E is transcoding, it changes the SSRC from the original RTP stream. Some phones do not respond well to an SSRC change in the middle of a call. This may occur when a phone changes from a CODEC that is passed through to a CODEC that is transcoded. (ID 7802)

Eyebeam Phones

Eyebeam softphones (Version 1.5.10.2 build 33793) do not play music-on hold generated by OS-E. (ID 7799)

Registration Plans and Registered States

In an LCS environment, a **registration-plan** must be configured to enable the location-cache to report an AOR in a registered state. Without a registration-plan, the AOR state will be declared as unregistered. The registration-plan must have a configured **route** with the **peer**, **action**, and **registration-throttling** settings configured. (For example, peer=LCS-server, action=tunnel, registration-throttling=no). (ID 8162)

CODEC Licensing

- If transcoding is configured with CODECs that require licenses and if a call is placed on hold without the available licenses, the call will be terminated. (ID 8206)
- In the OS-E Management System, some CODECs display a message saying they are "Available with upgrade." In fact, these CODECs are available, but the number of license seats, set at 200,000, is not configurable. This applies to the following CODECs: g728, g726-16, g726

SSH Session Limit Clarification

The SSH object **max-sessions** property sets the maximum number of concurrent SSH sessions allowed, enforced at the box level. The enforced value is an aggregate of the SSH session limits set on each IP interface that has SSH enabled. For example, to enforce a limit of five total SSH sessions per box, you could set IP "A" to an SSH session limit of two and IP "B" to an SSH session limit of three, for a total of five.

Call Failover

If you configure the fault-group so that a SIP process crash results in a VRRP failover, if the call-failover group is not set, and if the failed OS-E device is the call-failover master, the failover master-service is not transferred and the resulting call is lost. All active calls are deleted across all OS-E devices. (ID 7110)

Encryption of Fragmented RTP Packet

Currently, OS-E does not perform reassembly of fragmented RTP packets that require encryption. Because encryption requires the entire non-fragmented RTP packet, fragmented RTP packets are dropped. (ID 6462)

Eyebeam 1.5 Phone Disconnect

Performing Broadsoft session auditing on both legs of a SIP call will cause Eyebeam 1.5 phones running TLS/SRTP to disconnect. (ID 4700)

Rapid UDP Registrations and Maximum Sessions

When the **vsp\settings\max-number-of-sessions** property setting is reached, and with **vsp\call-admission-control** set to *disabled*, repeated error messages will occur if there are rapid UDP registrations consuming OS-E memory resources.

Use the **vsp\sip-timers\max-udp-session-linger** property to set the number of milliseconds that OS-E maintains a SIP session after its useful life is over. Enter a value from 0 to 60,000 milliseconds; the default setting is 30,000 ms (30 seconds). A value of 5 sets OS-E to remove the session 5 milliseconds after receipt/transmission of the first final response. (ID 5483)

NOTIFY Message From BroadSoft Server

If the **vsp\enterprise\servers\dns-group** is set with the **domain-port** property configured, or if the **local-port** is configured for an enterprise server, configure that port on all interfaces so that a SIP NOTIFY uses that port on the inbound call leg to the call destinations. Otherwise, the SIP NOTIFY will attempt to use a nonexisting local port and generate a “400 Bad Request” message back to the server on the trunk side of the network. (ID 5893)

OS-E Management System

- There will be a conflict the first time two users simultaneously open a OS-E configuration in the OS-E Management System (using the Configuration, Services or Access tab), and if one of the users does not make any changes or configuration updates before the other user. This problem will resolve itself after the first update without any actual configurations changes applied. However, there will be ongoing conflicts if the configuration contains a phantom directory. (ID 6182)
- The OS-E Management System is not supported over the Firefox Web browser. (ID 3506)
- In order for online help to display properly using Internet Explorer, you must be logged in to the OS-E Management System with an active session. The OS-E Management System is only supported over Internet Explorer, even though the default browser on your system may be different.

- After making any changes to the OS-E Web server configuration, such as changes made to the idle timeout and the connection protocol, and enabling and disabling the trap target, there may be a delay of three minutes while the Web server resets and allows the OS-E Management System to reconnect. Since there is a significant delay while the new settings are applied, the OS-E Management System user interface may be unresponsive if you attempt to perform other operations immediately after making changes to the Web server configuration. (ID 4052)

OS-E Actions Available at the NNOS-E> Prompt

The **set-call-forwarding** and **set-do-not-disturb** actions each show an optional cookie argument in the command line. The optional cookie argument is not supported for customer use. (ID 4180)

DHCP

DHCP is not currently implemented, even though you can configure DHCP using the OS-E Management System. (ID 4367)

Call Recording and Playback

- Recording of a video call will only record the audio portion of the call.
- After changing from an audio-only to audio & video call (and vice-versa), only the audio will be recorded.
- The call monitor-group "snoop" method does not function in this release. (ID 1100)
- Recorded calls cannot be played back using Windows Media Player V10.0 when the Web port is Port 443 (HTTPS). Windows Media Player only plays back recorded calls over Web port 80 (HTTP). As a workaround, use QuickTime instead of Windows Media Player, as QuickTime supports HTTPS. (ID 1732)

IM Management Policies

Message-to-sender and **message-to-recipient** text configured as part of an IM policy is not delivered as a separate IM message. The **message-to-recipient** text is prepended to the current message. The **message-to-sender** text is pre-pended to the next message going back to the sender. (ID 696)

Presence Database

In the **vsp/presence-database** object, the **repair-st-tags** and **force-un-subscribe** properties are available for debugging purposes only and are not intended for customer use. (ID 4618)

SMTP Archiving with Authentication

The password-tag and authentication functionality associated with SMTP archiving (**vsp\accounting\archiving\smtp-server**) is not currently supported. (ID 4521)

Policies

If you are using the header or content **sip-message-condition** properties, you must use the '(?s).' option at the beginning of the regular expression. For example, if you want to search for a user-agent RTC, use **`header match "(?s).*\bRTC/*\b".`** If you want to search for the media type in SDP, use **`content match "(?s).*\bm=video\b"`**

User and Group Filters

If you create a user or group filter under vsp/enterprise/directories, the filter will only take effect after a **directory-reset** action. (ID 2458)

LDAP and LDAP Authentication

- When you login to OS-E using SSH, it first asks you for a username and a password. This is the username/password combination of *root/sips* and not the LDAP username/password that you may have configured.
- Due to problems in the Java library, if an LDAP retrieval (import) of users fails, subsequent imports from other directories are not completed. This is a Java problem that will be fixed in a later JDK release. (ID 3300)

SNOM Phone Interoperability

- Certain random SRTP key values cause SNOM to play static or cause audio silence to the user. (ID 1587)
- SRTP with SNOM versions before v5.2 always produce static.

SRTP

Most RTCP implementations are still under development. If doing SRTP with an Eyebeam v1.5.6.1 or earlier, all RTCP packets requiring encryption (Sender and Receiver Reports) will be dropped. Since SNOM phones do not include authentication in RTCP packets, all SNOM RTCP Sender and Receiver Report packets are dropped. (ID 5206).

Linksys SRTP

In the **vsp/default-session-config/media** object, if you set the **rtcp** property to *pass*, OS-E will not send RTCP packets to endpoints in a Linksys SRTP configuration. (ID 6878)

Registration-Plan

The **location-service settings** object and the **registration-plan route** object both allow you to set the **max-bindings-per-AOR** property. Currently this value is set to 1 in settings and 0 (as-is) in route, and should not be changed.

Archiving

- If OS-E is configured for registration delegation (instead of handling registrations directly), and if archiving is enabled, SIP calls will be archived twice, with both archives containing the same information. (ID 2658)
- When configuring archiving based on the new **record-count** property in Release 3.1, calls made through a server (such as Asterisk) originate two records per call so that the number of records is twice the real number of calls. For example, if you configure the archive **record-count** to 100, archiving occurs at the 50th call because of the two records per call. (ID 5335)

- If using the **record-count** property to set the threshold that determines the number of accounting records to be written before the whole group is then archived, setting a value to 0 (the default), disables this feature. When set to any number greater than 0, the accounting software checks every 30 seconds to calculate whether the requisite number of records are waiting.

  Archiving only occurs when you invoke it specifically, as an action or as a scheduled text. Note that setting this property to any value other than 0 causes both the archive action and/or task to fail. Use this property with care, as the archiving function consumes system resources. (ID 5387, 5404)