



CLI Configuration Guide for Cisco Unified SIP Proxy Release 9.0

October 10, 2019

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: The equipment described in this manual generates and may radiate radio-frequency energy. If it is not installed in accordance with Cisco's installation instructions, it may cause interference with radio and television reception. This equipment has been tested and found to comply with the limits for a Class B digital device in accordance with the specifications in part 15 of the FCC rules. These specifications are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

Modifying the equipment without Cisco's written authorization may result in the equipment no longer complying with FCC requirements for Class A or Class B digital devices. In that event, your right to use the equipment may be limited by FCC regulations, and you may be required to correct any interference to radio or television communications at your own expense.

You can determine whether your equipment is causing interference by turning it off. If the interference stops, it was probably caused by the Cisco equipment or one of its peripheral devices. If the equipment causes interference to radio or television reception, try to correct the interference by using one or more of the following measures:

- Turn the television or radio antenna until the interference stops.
- Move the equipment to one side or the other of the television or radio.
- Move the equipment farther away from the television or radio.
- Plug the equipment into an outlet that is on a different circuit from the television or radio. (That is, make certain the equipment and the television or radio are on circuits controlled by different circuit breakers or fuses.)

Modifications to this product not authorized by Cisco Systems, Inc. could void the FCC approval and negate your authority to operate the product.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

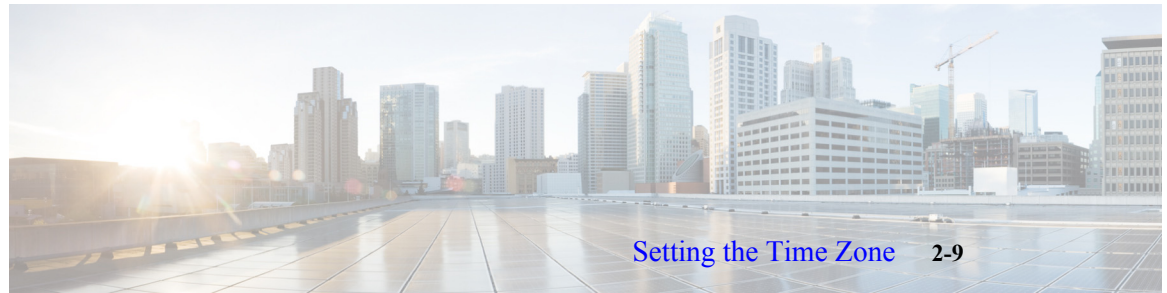
Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

CLI Configuration Guide for Cisco Unified SIP Proxy Release 9.0

© 2019 Cisco Systems, Inc. All rights reserved.



Overview of Cisco Unified SIP Proxy Release 9.0	1-1
About This Document	1-1
Administration Interfaces	1-1
Command-Line Interface	1-1
Graphical User Interface	1-2
Commercial Open Source Licensing	1-2
Obtaining Documentation and Submitting a Service Request	1-2
Technical Assistance	1-2
Initial Configuration Tasks	2-1
Configuring Smart Licensing	2-1
About Smart Licensing	2-1
Summary Steps	2-1
Detailed Steps	2-2
Example	2-2
Setting Backup Parameters	2-3
About Backup Parameters	2-3
Prerequisites	2-3
Summary Steps	2-3
Detailed Steps	2-3
Example	2-4
Configuring NTP Servers	2-5
Adding NTP Servers	2-5
About Adding NTP Servers	2-5
Summary Steps	2-5
Detailed Steps	2-6
Examples of Adding NTP Servers	2-6
Removing an NTP Server	2-7
Summary Steps	2-7
Detailed Steps	2-7
Displaying NTP Server Information	2-8
Commands to Display NTP Server Information	2-8
Examples of Showing NTP Server Information	2-8



Setting the Time Zone 2-9

Example of Setting the Time Zone 2-9

Configuring HTTPS for Administration Web Interface 2-10

Summary Steps 2-10

Detailed Steps 2-11

Example of Configuring HTTPS 2-11

Configuring the Cisco Unified SIP Proxy 3-1

Configuring Logical Networks 3-1

Summary Steps 3-1

Detailed Steps 3-2

Example 3-2

Configuring Trigger Conditions 3-2

Summary Steps 3-3

Detailed Steps 3-3

Example 3-4

Configuring Server Groups 3-4

About Server Groups 3-4

Summary Steps 3-5

Detailed Steps 3-5

Example 3-6

Configuring Route Tables 3-6

About Route Tables 3-6

Summary Steps 3-6

Detailed Steps 3-7

Example 3-7

Configuring Normalization Policies 3-8

Summary Steps 3-8

Detailed Steps 3-8

Example 3-9

Configuring Lookup Policies 3-9

Summary Steps 3-9

Detailed Steps 3-10

Example 3-10

Configuring Routing Triggers	3-11
Summary Steps	3-11
Detailed Steps	3-11
Example	3-11
Configuring Normalization Triggers	3-12
Summary Steps	3-12
Detailed Steps	3-12
Example	3-13
Configuring Listen and Record-Route Ports	3-13
Summary Steps	3-13
Detailed Steps	3-13
Example	3-14
Configuring a Hostname	3-14
Summary Steps	3-14
Detailed Steps	3-15
Example	3-15
Configuring Transport Layer Security (TLS)	3-15
Creating and Importing a Signed Certificate	3-15
Prerequisites	3-15
Summary Steps	3-16
Detailed Steps	3-16
Example of Creating a Signed Certificate	3-16
Configuring TLS on Cisco Unified SIP Proxy	3-17
Summary Steps	3-17
Detailed Steps	3-18
Example of Configuring TLS	3-18
Configuring Lite Mode	3-18
Summary Steps	3-19
Detailed Steps	3-19
Example	3-19
Configuring Performance Control	3-19
About Performance Control	3-20
Summary Steps	3-20
Detailed Steps	3-20
Example	3-20
Committing the Configuration	3-20
Backing Up and Restoring Data	4-1
About Backing Up and Restoring Data	4-1

Restrictions for Backing Up and Restoring Data	4-1
Backing Up Files	4-2
About Backing Up Files	4-2
Summary Steps	4-2
Detailed Steps	4-3
Examples	4-3
Restoring Files	4-4
About Restoring Files	4-4
Summary Steps	4-4
Detailed Steps	4-5
Related Topics	4-5
Maintaining the Cisco Unified SIP Proxy System	5-1
Copying Configurations	5-1
Copying the Startup Configuration from the Hard Disk to Another Location	5-1
Copying the Startup Configuration from the Network FTP Server to Another Location	5-2
Copying the Running Configuration from the Hard Disk to Another Location	5-2
Copying the Running Configuration from the Network TFTP Server to Another Location	5-3
Checking Hard Disk Memory Wear Activity	5-3
Troubleshooting	6-1
Using CLI Commands to Troubleshoot the System	6-1
About Logging	6-1
Log Commands	6-2
Example of Log Output	6-2
Using Trace Commands	6-2
Using Show Commands	6-3
Troubleshooting Configuration Changes	6-3
Related Topics	6-3
Configuration Example	7-1



Overview of Cisco Unified SIP Proxy Release 9.0

Last updated: October 10, 2019

- [About This Document, page 1](#)
- [Administration Interfaces, page 1](#)
- [Commercial Open Source Licensing, page 2](#)
- [Obtaining Documentation and Submitting a Service Request, page 2](#)
- [Technical Assistance, page 2](#)

About This Document

This document contains information about how to configure the Cisco Unified SIP Proxy system using the CLI. Use it in conjunction with the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#), which lists all the CLI commands.

Administration Interfaces

Cisco Unified SIP Proxy Release 9.0 utilizes both a command-line interface (CLI) and a graphical user interface (GUI).

- [Command-Line Interface, page 1](#)
- [Graphical User Interface, page 2](#)

Command-Line Interface

The CLI is a text-based interface accessed through a Telnet session to the router hosting Cisco Unified SIP Proxy. Those familiar with Cisco IOS command structure and routers will see similarities.

The Cisco Unified SIP Proxy commands are structured much like the Cisco IOS CLI commands. However, the Cisco Unified SIP Proxy CLI commands do not affect Cisco IOS configurations. After you log in to Cisco Unified SIP Proxy, the command environment is no longer the Cisco IOS environment.

The CLI is accessible from a PC or server anywhere in the IP network.

Graphical User Interface

Cisco Unified SIP Proxy Release 9.0 introduces a GUI that is used to configure and operate the Cisco Unified SIP Proxy system.

For information on using the GUI, see the online help in the application or the *GUI Administration Guide for Cisco Unified SIP Proxy Release 9.0*.

GUI information is not within the scope of this document.

Commercial Open Source Licensing

Some components of the software created for Cisco Unified SIP Proxy Release 9.0 are provided through open source or commercial licensing. These components and the associated copyright statements can be found at http://www.cisco.com/en/US/products/ps10475/products_licensing_information_listing.html.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and RSS Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com username and password.</p>	http://www.cisco.com/techsupport
<p>Use the Cisco Feature Navigator website to find information about platform support and Cisco IOS and Catalyst OS software image support. An account on Cisco.com is not required.</p>	http://www.cisco.com/go/cfn



Initial Configuration Tasks

Last updated: October 10, 2019

- [Configuring Smart Licensing, page 1](#)
- [Setting Backup Parameters, page 3](#)
- [Configuring NTP Servers, page 5](#)
- [Setting the Time Zone, page 9](#)
- [Configuring HTTPS for Administration Web Interface, page 10](#)

Configuring Smart Licensing

- [About Smart Licensing, page 1](#)
- [Summary Steps, page 1](#)
- [Detailed Steps, page 2](#)
- [Example, page 2](#)

About Smart Licensing

Cisco Smart Software Licensing is a standardized licensing platform that facilitates you to deploy and manage Cisco software licenses easily and quickly. Cisco Smart Software Licensing establishes a pool of software licenses that can be used across your network in a flexible and automated manner. It also provides visibility to your purchased and deployed licenses in your network. Cisco Smart Software Licensing removes the need for Product Activation Keys (PAKs) and reduces your license activation and registration time.

Summary Steps

1. **enable**
2. **license smart destinationAddr *url***
3. **license smart httpProxyAddr *url***
4. **license smart activate cusp *count***
5. **license smart register token_id *token***

Detailed Steps

	Command or Action	Purpose
Step 1	enable Example: se-10-0-0-0# enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	license smart destinationAddr https://tools.cisco.com/its/service/oddce/services/DDCEService Example: se-10-0-0-0# license smart destinationAddr https://tools.cisco.com/its/service/oddce/services/DDCEService	Connects to the central licensing server.
Step 3	license smart httpProxyAddr 10.1.1.1 Example: se-10-0-0-0# license smart httpProxyAddr 10.1.1.1	Sets the HTTP(S) proxy server address for smart licensing.
Step 4	license smart activate cusp count Example: se-10-0-0-0# license smart activate cusp 100	Activates the request number of licenses. The count should be multiple of 5.
Step 5	license smart register token_id token Example: se-10-0-0-0# license smart register token_id MjgxZjdY2RtMWY5Ny00YTk4LOI2N2MtNjcxcNmYaMTkzZGFhLHE0MjA3MjY0%0AMjI5N34Z8OVAOdmNzSjdIeG4MMHIzTmZubNFzMHhKOTYyeHlUZWQzQzVIM3Jk%0AHV3MD0A3D%0N	Registers the device instance with the Cisco licensing cloud. This step needs to be performed only once per device instance. The license agent registers the product with Cisco and receives back an identity certificate. This certificate is saved and automatically used for all future communications with Cisco. The license agent automatically renews the registration information with Cisco after one year.

Example

The following example configures Smart License on the Cisco Unified SIP Proxy:

```
se-10-0-0-0# enable
se-10-0-0-0# license smart destinationAddr
https://tools.cisco.com/its/service/oddce/services/DDCEService
se-10-0-0-0# license smart httpProxyAddr 10.1.1.1
se-10-0-0-0# license smart activate cusp 100
se-10-0-0-0# license smart register token_id
MjgxZjdY2RtMWY5Ny00YTk4LOI2N2MtNjcxcNmYaMTkzZGFhLHE0MjA3MjY0%0AMjI5NDZ8OVAOdmNzSjdIeG4MMHIzTmZubNFzMHhKOTYyeHl67ZWQzQzVIM3Jk%0AHV3MD0A3D%0N
```

Setting Backup Parameters

- [About Backup Parameters, page 3](#)
- [Prerequisites, page 3](#)
- [Summary Steps, page 3](#)
- [Detailed Steps, page 3](#)
- [Example, page 4](#)

About Backup Parameters

Cisco Unified SIP Proxy backup and restore functions use an FTP server to store and retrieve data. The backup function copies the files from Cisco Unified SIP Proxy to the FTP server and the restore function copies the files from the FTP server to Cisco Unified SIP Proxy. The FTP server can reside anywhere in the network as long as the backup and restore functions can access it with an IP address or hostname.

All Cisco Unified SIP Proxy backup files are stored on the specified server. You can copy the backup files to other locations or servers, if necessary.

The backup parameters specify the FTP server to use for storing Cisco Unified SIP Proxy backup files and the number of backups that are stored before the system overwrites the oldest one.

Prerequisites

- Verify that an FTP administrator or other user who can log in to the FTP server has full permission on the FTP server, such as read, write, overwrite, create, and delete permissions for files and directories.
- Gather the FTP server URL and the username and password of the FTP server login.
- Determine the number of revisions to save before the oldest backup is overwritten.

Summary Steps

6. **configure terminal**
7. **backup server url** *backup-ftp-url* **username** *backup-ftp-username* **password** *backup-ftp-password*
8. **backup revisions number** *number*
9. **end**
10. **show backup**

Detailed Steps

	Command or Action	Purpose
Step 1	configure terminal	Enters configuration mode.
	<i>se-10-0-0-0#</i> config terminal	

	Command or Action	Purpose
Step 2	<pre>backup server url ftp-url username ftp-username password ftp-password} Example: se-10-0-0-0(config)> backup server url ftp://main/backups username "admin" password "wxyz" se-10-0-0-0(config)> backup server url ftp://192.0.2.15/backups username "admin" password "wxyz"</pre>	<p>Sets the backup parameters.</p> <p>Note You must configure the backup server before you can configure the backup revisions.</p> <ul style="list-style-type: none"> server url—The <i>ftp-url</i> value is the URL to the network FTP server where the backup files will be stored. The <i>ftp-username</i> and <i>ftp-password</i> values are the username and password for the network FTP server. <p>In the example, main is the hostname of the FTP server and backups is the directory where backup files are stored.</p>
Step 3	<pre>backup revisions number Example: se-10-0-0-0(config)> backup revisions 5</pre>	<p>Sets the number of backup files that will be stored. When the system reaches this number of backups, it deletes the oldest stored file.</p>
Step 4	<pre>end Example: se-10-0-0-0(config)> end</pre>	<p>Exits configuration mode.</p>
Step 5	<pre>show backup Example: se-10-0-0-0> show backup</pre>	<p>Displays the backup server configuration information, including the FTP server URL and the maximum number of backup files available.</p>

Example

The following example configures a backup server and displays the **show backup** output:

```
se-10-0-0-0> enable
se-10-0-0-0# configure terminal
se-10-0-0-0(config)> backup revisions 5
se-10-0-0-0(config)> backup server url ftp://10.12.0.1/ftp username "admin" password
"wxyz"
se-10-0-0-0(config)> end
se-10-0-0-0> show backup
Server URL:                               ftp://10.12.0.1/ftp
User Account on Server:
Number of Backups to Retain:              5
se-10-0-0-0>
```

Related Topics

- For information about the CLI commands, see the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#).
- For information about backing up and restoring your configuration, see [Backing Up and Restoring Data](#).

Configuring NTP Servers

When you install the Cisco Unified SIP Proxy software, the system gives you the option of adding up to two Network Time Protocol (NTP) servers. You can add additional NTP servers (the system supports up to three NTP servers), remove one or more NTP servers, or display NTP server information using the CLI.

- [Adding NTP Servers, page 5](#)
- [Removing an NTP Server, page 7](#)
- [Displaying NTP Server Information, page 8](#)

Adding NTP Servers

- [About Adding NTP Servers, page 5](#)
- [Summary Steps, page 5](#)
- [Detailed Steps, page 6](#)
- [Examples of Adding NTP Servers, page 6](#)

About Adding NTP Servers

You can specify an NTP server using its IP address or its hostname.

Cisco Unified SIP Proxy uses the DNS server to resolve the hostname to an IP address and stores the IP address as an NTP server. If DNS resolves the hostname to more than one IP address, Cisco Unified SIP Proxy randomly chooses one of the IP addresses that is not already designated as an NTP server. If you do not want to go with the random choice, set the **prefer** attribute for one server.

To configure an NTP server with multiple IP addresses for a hostname, repeat the configuration steps using the same hostname. Each iteration assigns the NTP server to its remaining IP addresses.

Summary Steps

1. **configure terminal**
2. **ntp server {hostname | ip-address} [prefer]**
3. **end**
4. **show ntp status**
5. **show ntp configuration**
6. **copy running-config startup-config**

Detailed Steps

	Command or Action	Purpose
Step 1	<code>configure terminal</code> Example: <code>se-10-0-0-0# config terminal</code>	Enters configuration mode.
Step 2	<code>ntp server {hostname ip-address} [prefer]</code> Example: <code>se-10-0-0-0(config) > ntp server 192.0.2.14</code> <code>se-10-0-0-0(config) > ntp server 192.0.2.17 prefer</code>	Specifies the hostname or IP address of the NTP server. If more than one server is configured, the server with the prefer attribute is used before the others.
Step 3	<code>end</code> Example: <code>se-10-0-0-0(config) > exit</code>	Exits configuration mode.
Step 4	<code>show ntp status</code> Example: <code>se-10-0-0-0> show ntp status</code>	Displays statistics about the NTP server.
Step 5	<code>show ntp configuration</code> Example: <code>se-10-0-0-0> show ntp configuration</code>	Displays the configured NTP servers.
Step 6	<code>copy running-config startup-config</code> Example: <code>se-10-0-0-0> copy running-config startup-config</code>	Copies the configuration changes to the startup configuration.

Examples of Adding NTP Servers

The following commands configure the NTP server:

```
se-10-0-0-0# configure terminal
se-10-0-0-0(config)> ntp server 192.0.2.14
se-10-0-0-0(config)> exit
se-10-0-0-0>
```

The output from the **show ntp status** command looks similar to the following:

```
se-10-0-0-0> show ntp status

NTP reference server 1:      192.0.2.14
Status:                     sys.peer
Time difference (secs):     3.268110099434328E8
Time jitter (secs):         0.1719226837158203
```

Removing an NTP Server

You can remove an NTP server using its IP address or hostname.

- [Summary Steps, page 7](#)
- [Detailed Steps, page 7](#)

Summary Steps

1. **configure terminal**
2. **no ntp server {hostname | ip-address}**
3. **exit**
4. **show ntp status**
5. **show ntp configuration**
6. **copy running-config startup-config**

Detailed Steps

	Command or Action	Purpose
Step 1	configure terminal Example: se-10-0-0-0# configure terminal	Enters configuration mode.
Step 2	no ntp server {hostname ip-address} Example: se-10-0-0-0(config)> no ntp server 192.0.2.14 se-10-0-0-0(config)> no ntp server myhost	Specifies the hostname or IP address of the NTP server to remove.
Step 3	exit Example: se-10-0-0-0(config)> exit	Exits configuration mode.
Step 4	show ntp status Example: se-10-0-0-0> show ntp status	Displays statistics about the NTP server.
Step 5	show ntp configuration Example: se-10-0-0-0> show ntp status	Displays the configured NTP servers.
Step 6	copy running-config startup-config Example: se-10-0-0-0> copy running-config startup-config	Copies the configuration changes to the startup configuration.

Displaying NTP Server Information

- [Commands to Display NTP Server Information, page 8](#)
- [Examples of Showing NTP Server Information, page 8](#)

Commands to Display NTP Server Information

The following commands are available to display NTP server configuration information and status:

- **show ntp associations**
- **show ntp servers**
- **show ntp source**
- **show ntp status**

Examples of Showing NTP Server Information

The following is sample output for the **show ntp associations** command:

```
se-10-0-0-0> show ntp associations

ind assID status  conf reach auth condition  last_event cnt
=====
   1 61253  8000   yes   yes  none    reject
```

The following is sample output for the **show ntp servers** command:

```
se-10-0-0-0> show ntp servers

      remote          refid      st t when poll reach  delay  offset  jitter
=====
  1.100.6.9          0.0.0.0          16 u   - 1024    0   0.000   0.000 4000.00
space reject,        x falsetick,      . excess,        - outlier
+ candidate,         # selected,      * sys.peer,      o pps.peer
```

The following is sample output for the **show ntp source** command:

```
se-10-0-0-0> show ntp source

127.0.0.1: stratum 16, offset 0.000013, synch distance 8.67201
0.0.0.0:      *Not Synchronized*
```

The following is sample output for the **show ntp status** command:

```
se-10-0-0-0> show ntp status

NTP reference server :      10.100.6.9
Status:                reject
Time difference (secs):    0.0
Time jitter (secs):       4.0
```

Related Topics

- For information about the CLI commands, see the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#).
- For information about the initial installation of the Cisco Unified SIP Proxy system and the post installation configuration tool, see the [Installation Guide for Cisco Unified SIP Proxy Release 9.0](#).
- For information about copying the configuration, see [Copying Configurations, page 1](#).

Setting the Time Zone

When you install the Cisco Unified SIP Proxy software, the system prompts you to set the time zone. If you need to change it, use the **clock timezone** command in Cisco Unified SIP Proxy configuration mode.

To display the time zone, use the **show clock detail** command in module EXEC mode.

Example of Setting the Time Zone

```
se-10-0-0-0# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
se-10-0-0-0(config)# clock timezone
Please identify a location so that time zone rules can be set correctly.
Please select a continent or ocean.
1) Africa                4) Arctic Ocean          7) Australia            10) Pacific Ocean
2) Americas              5) Asia                  8) Europe
3) Antarctica            6) Atlantic Ocean        9) Indian Ocean
>? 2
Please select a country.
1) Anguilla              18) Ecuador              35) Paraguay
2) Antigua & Barbuda     19) El Salvador          36) Peru
3) Argentina             20) French Guiana        37) Puerto Rico
4) Aruba                 21) Greenland            38) St Kitts & Nevis
5) Bahamas              22) Grenada              39) St Lucia
6) Barbados              23) Guadeloupe           40) St Pierre & Miquelon
7) Belize               24) Guatemala            41) St Vincent
8) Bolivia               25) Guyana               42) Suriname
9) Brazil               26) Haiti                43) Trinidad & Tobago
10) Canada              27) Honduras            44) Turks & Caicos Is
11) Cayman Islands      28) Jamaica              45) United States
12) Chile               29) Martinique           46) Uruguay
13) Colombia            30) Mexico               47) Venezuela
14) Costa Rica          31) Montserrat           48) Virgin Islands (UK)
15) Cuba               32) Netherlands Antilles 49) Virgin Islands (US)
16) Dominica            33) Nicaragua
17) Dominican Republic 34) Panama
>? 45
Please select one of the following time zone regions.
1) Eastern Time
2) Eastern Time - Michigan - most locations
3) Eastern Time - Kentucky - Louisville area
4) Eastern Time - Kentucky - Wayne County
5) Eastern Standard Time - Indiana - most locations
6) Eastern Standard Time - Indiana - Crawford County
7) Eastern Standard Time - Indiana - Starke County
8) Eastern Standard Time - Indiana - Switzerland County
9) Central Time
10) Central Time - Michigan - Wisconsin border
11) Central Time - North Dakota - Oliver County
12) Mountain Time
13) Mountain Time - south Idaho & east Oregon
14) Mountain Time - Navajo
15) Mountain Standard Time - Arizona
16) Pacific Time
17) Alaska Time
18) Alaska Time - Alaska panhandle
19) Alaska Time - Alaska panhandle neck
20) Alaska Time - west Alaska
21) Aleutian Islands
22) Hawaii
```

```
>? 16
```

The following information has been given:

```
United States  
Pacific Time
```

```
Therefore TZ='America/Los_Angeles' will be used.  
Local time is now:      Mon Aug 27 17:23:54 PDT 2007.  
Universal Time is now:  Tue Aug 28 00:23:54 UTC 2007.  
Is the above information OK?  
1) Yes  
2) No  
>? 1
```

Save the change to startup configuration and reload the module for the new time zone to take effect.

```
se-10-0-0-0(config)>
```

Configuring HTTPS for Administration Web Interface

You can configure the system to allow HTTPS access to Cisco Unified SIP Proxy GUI.

- [Summary Steps, page 10](#)
- [Detailed Steps, page 11](#)
- [Example of Configuring HTTPS, page 11](#)

Summary Steps

1. **configure**
2. **crypto key generate rsa label *labelname* modulus 1024**
3. **web session security keylabel *labelname***
4. **end**

Detailed Steps

	Command or Action	Purpose
Step 1	configure Example: se-10-0-0-0> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 2	crypto key generate rsa label <i>labelname</i> modulus 1024 Example: se-10-0-0-0(cusp-config)> crypto key generate rsa label mykey modulus 1024 Key generation in progress. Please wait... The label name for key is mykey	Generates a self-signed certificate and an RSA private key.
Step 3	web session security keylabel <i>labelname</i> Example: se-10-0-0-0(cusp-config)> web session security keylabel mykey	Associates a security key for HTTPS.
Step 4	end Example: se-10-0-0-0(cusp-config)> end	Exits to privileged EXEC mode.

Example of Configuring HTTPS

```

se-10-0-0-0> configure
se-10-0-0-0(cusp)> crypto key generate rsa label mykey modulus 1024
se-10-0-0-0(cusp-config)> web session security keylabel mykey
se-10-0-0-0(cusp-config)> end

```




Configuring the Cisco Unified SIP Proxy

Last updated: October 10, 2019

- [Configuring Logical Networks, page 1](#)
- [Configuring Trigger Conditions, page 2](#)
- [Configuring Server Groups, page 4](#)
- [Configuring Route Tables, page 6](#)
- [Configuring Normalization Policies, page 8](#)
- [Configuring Lookup Policies, page 9](#)
- [Configuring Routing Triggers, page 11](#)
- [Configuring Normalization Triggers, page 12](#)
- [Configuring Listen and Record-Route Ports, page 13](#)
- [Configuring a Hostname, page 14](#)
- [Configuring Transport Layer Security \(TLS\), page 15](#)
- [Configuring Lite Mode, page 18](#)
- [Configuring Performance Control, page 19](#)
- [Committing the Configuration, page 20](#)

Configuring Logical Networks

Each interface on the Cisco Unified SIP Proxy is associated with a logical network. Logical networks are used to organize server groups, listen points, and other properties. SIP messages are associated with the network on which they arrive.

- [Summary Steps, page 1](#)
- [Detailed Steps, page 2](#)
- [Example, page 2](#)

Summary Steps

1. `cusp`

- 2. **configure**
- 3. **sip network** *network*
- 4. **end network**

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp) > configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	sip network <i>network</i> Example: se-10-0-0-0(cusp-config) > sip network service-provider	Creates a network and puts you into network command mode. In this case, the network that is being created is called “service provider”.
Step 4	end network Example: se-10-0-0-0(cusp-config-network) > end network	Exits network command mode.

Example

The following example creates a network called “service-provider”:

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp) > configure
se-10-0-0-0(cusp-config) > sip network service-provider
se-10-0-0-0(cusp-config-network) > end network
```

Configuring Trigger Conditions

You create trigger conditions to allow Cisco Unified SIP Proxy to respond with the appropriate action for various call flows. In general, the more complex the call flow is, the more complex the trigger must be.

- [Summary Steps, page 3](#)
- [Detailed Steps, page 3](#)
- [Example, page 4](#)

Summary Steps

1. **cusp**
2. **configure**
3. **trigger condition** *trigger-condition-name*
4. **sequence** *sequence-number*
5. (Optional) **in-network** *network-name*
6. (Optional) **mid-dialog**
7. **end sequence**
8. **end trigger condition**

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	trigger condition <i>trigger-condition-name</i> Example: se-10-0-0-0(cusp-config)> trigger condition call-from-service-provider	Creates a trigger condition and puts you into trigger command mode. In this case, the trigger that is being created is called “call-from-service-provider”.
Step 4	sequence <i>sequence-number</i> Example: se-10-0-0-0(cusp-config-trigger)> sequence 1	Creates a sequence with the specified number and puts you into trigger sequence command mode. The number indicates the order in which triggers are evaluated. In this case, the sequence that is being created is sequence number 1.
Step 5	in-network <i>network-name</i> Example: se-10-0-0-0(cusp-config-trigger-seq)> in-network service-provider	Optional. Specifies the incoming network name for the trigger condition. In this case, the incoming network is the “service-provider” network.
Step 6	mid-dialog Example: se-10-0-0-0(cusp-config-trigger-seq)> mid-dialog	Optional. A special trigger that bypasses routing policies on mid-dialog messages.

	Command or Action	Purpose
Step 7	end sequence Example: se-10-0-0-0(cusp-config-trigger-seq) > end sequence	Exits the trigger sequence command mode.
Step 8	end trigger condition Example: se-10-0-0-0(cusp-config-trigger) > end trigger condition	Exits the trigger command mode.

Example

In this example, Cisco Unified SIP Proxy only reacts based on the network the call came in on, so the triggers are simple.

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> trigger condition call-from-service-provider
se-10-0-0-0(cusp-config-trigger)> sequence 1
se-10-0-0-0(cusp-config-trigger-seq)> in-network service-provider
se-10-0-0-0(cusp-config-trigger-seq)> end sequence
se-10-0-0-0(cusp-config-trigger)> end trigger condition

se-10-0-0-0(cusp-config)> trigger condition mid-dialog
se-10-0-0-0(cusp-config-trigger)> sequence 1
se-10-0-0-0(cusp-config-trigger-seq)> mid-dialog
se-10-0-0-0(cusp-config-trigger-seq)> end sequence
se-10-0-0-0(cusp-config-trigger)> end trigger condition
```

Configuring Server Groups

- [About Server Groups, page 4](#)
- [Summary Steps, page 5](#)
- [Detailed Steps, page 5](#)
- [Example, page 6](#)

About Server Groups

Server groups define the elements that Cisco Unified SIP Proxy interacts with for each network. The server group name that is used is inserted into the SIP URI of the outgoing request. Some devices, such as Cisco Unified Communications Manager, validate the URI of requests before processing, which means that the end device might need to be configured with a Fully Qualified Domain Name (FQDN) to allow for this.

Two of the fields for each individual element, q-value and weight, are important to use to specify the priorities of elements, and also for load balancing. Calls are routed to specific elements based on q-value. The element with the highest q-value receives all traffic routed to that server group. If multiple elements have the same q-value, traffic is distributed between them based on the load-balancing option used. The

default load-balancing is based on call-id, but weight can also be used. If weight is used, the percentage of traffic that an element receives is equal to its weight divided by the sum of up elements with the same q-value's weights. The sum of their weights does not need to equal 100. You can change the weights and q-values to configure a different priority or load-balancing scheme.

Summary Steps

1. **cusp**
2. **configure**
3. **server-group sip group *server-group-name network***
4. **element ip-address *ipaddress port {udp | tcp | tls} [q-value *q-value*] [weight *weight*]***
5. **lb-type {global | highest-q | request-uri | call-id | to-uri | weight }**
6. **end server-group**

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	server-group sip group <i>server-group-name network</i> Example: se-10-0-0-0(cusp-config)> server-group sip group sp.example.com service-provider	Creates a SIP server group and enters server group command mode. In this case, the server group being created is called “sp.example.com” and it uses the network called “service-provider”.
Step 4	element ip-address <i>ipaddress port {udp tcp tls} [q-value <i>q-value</i>] [weight <i>weight</i>]</i> Example: se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.3 5060 tls q-value 1.0 weight 100	Creates an IP element for a SIP server group and determines the characteristics of the SIP server group. Note You can enter this command multiple times.

	Command or Action	Purpose
Step 5	lb-type { global highest-q request-uri call-id to-uri weight } Example: se-10-0-0-0(cusp-config-sg) > lb-type weight	Configures the load-balancing algorithm for the SIP server group. In this example, it specifies that the element will be selected proportional to its weight relative to the weights of other elements of the same q-value.
Step 6	end server-group Example: se-10-0-0-0(cusp-config-sg) > end server-group	Exits the server group command mode.

Example

```

se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> server-group sip group sp.example.com service-provider
se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.3 5060 tls q-value 1.0 weight 100
se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.4 5060 tls q-value 1.0 weight 50
se-10-0-0-0(cusp-config-sg)> element ip-address 192.168.10.5 5060 tls q-value 1.0 weight 50
se-10-0-0-0(cusp-config-sg)> lb-type weight
se-10-0-0-0(cusp-config-sg)> end server-group

```

Configuring Route Tables

- [About Route Tables, page 6](#)
- [Summary Steps, page 6](#)
- [Detailed Steps, page 7](#)
- [Example, page 7](#)

About Route Tables

You must configure route tables to direct SIP requests to their appropriate destinations. Each route table consists of a set of keys that are matched based on the lookup policy. For example, each key might represent the prefix of a phone number dialed.

Summary Steps

1. **cusp**
2. **configure**
3. **route table** *table-name*
4. **key** *key* **response** *response-code*
5. **key** *key* **target-destination** *target-destination network*

6. end route table

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	route table table-name Example: se-10-0-0-0(cusp-config)> route table service-provider-table	Creates a route table and enters route table command mode. In this case, it creates a route table called “service-provider-table”.
Step 4	key key response response-code Example: se-10-0-0-0(cusp-config-rt)> key * response 404	Assigns a response code to a lookup key. In this example, it returns a response of “404” to everything.
Step 5	key key target-destination target-destination network Example: se-10-0-0-0(cusp-config-rt)> key 510 target-destination cube-sp.example.com cube-sp	Replaces the key part of the target destination with a specified value. Note You can enter this command multiple times.
Step 6	end route table Example: se-10-0-0-0(cusp-config-rt)> end route table	Exits the route table command mode.

Example

```

se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> route table service-provider-table
se-10-0-0-0(cusp-config-rt)> key * response 404
se-10-0-0-0(cusp-config-rt)> key 510 target-destination cube-sp.example.com cube-sp
se-10-0-0-0(cusp-config-rt)> end route table

```

Configuring Normalization Policies

Normalization policies modify SIP messages to account for incompatibilities between networks. In this case, the service provider cannot handle phone numbers with the escape sequence “91,” so the sequence must be removed from the request-uri and TO header.

- [Summary Steps, page 8](#)
- [Detailed Steps, page 8](#)
- [Example, page 9](#)

Summary Steps

1. **cusps**
2. **configure**
3. **policy normalization** *policy_name*
4. **uri-component update request-uri** {user | host | host-port | phone | uri} {all | match-string} *replace-string*
5. **uri-component update header** {first | last | all} {user | host | host-port | phone | uri} {all | match-string} *replace-string*
6. **end policy**

Detailed Steps

	Command or Action	Purpose
Step 1	cusps Example: se-10-0-0-0> cusps	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusps)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	policy normalization <i>policy-name</i> Example: se-10-0-0-0(cusps-config)> policy normalization outgoing-norm-policy	Creates a normalization policy and enters policy normalization command mode. In this example, the normalization policy is called “outgoing-norm-policy”.
Step 4	uri-component update request-uri {user host host-port phone uri} {all match-string} <i>replace-string</i> Example: se-10-0-0-0(cusps-config-norm)> uri-component update request-uri user ^91 ""	Configures a normalization policy step that updates a URI component field within a request URI.

	Command or Action	Purpose
Step 5	<pre>uri-component update header {first last all} {user host host-port phone uri} {all match-string} replace-string</pre> <p>Example: se-10-0-0-0(cusp-config-norm) > uri-component update TO all user ^91 ""</p>	Configures a normalization policy step that updates a URI component field within a header of the source message.
Step 6	<pre>end policy</pre> <p>Example: se-10-0-0-0(cusp-config-norm) > end policy</p>	Exits policy normalization command mode.

Example

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp) > configure
se-10-0-0-0(cusp-config) > policy normalization outgoing-norm-policy
se-10-0-0-0(cusp-config-norm) > uri-component update request-uri user ^91 ""
se-10-0-0-0(cusp-config-norm) > uri-component update TO all user ^91 ""
se-10-0-0-0(cusp-config-norm) > end policy
```

Configuring Lookup Policies

Lookup policies decide how the keys in the route tables are used. Each key represents the beginning of the phone number dialed because each policy states to match the user component of the request-uri against the keys in its route table. The user component of the request-uri is the phone number called. The rule used to match is prefix, which means that the longest prefix match in the route table is used. So if the dialed number is 510-1XX-XXXX, the call is sent to the cme.example.com server group. If the dialed number is 510-XXX-XXXX, the call is sent to the cucm.example.com server group. The four policies in the following example are identical, except that they each refer to their specific table.

- [Summary Steps, page 9](#)
- [Detailed Steps, page 10](#)
- [Example, page 10](#)

Summary Steps

1. **cusp**
2. **configure**
3. **policy lookup** *policy-name*
4. **sequence** *sequence-number*
5. **rule** {exact | prefix | subdomain | subnet | fixed length} [case-insensitive]
6. **end sequence**
7. **end policy**

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	policy lookup <i>policy-name</i> Example: se-10-0-0-0(cusp-config)> policy lookup service-provider-policy	Creates a policy with the specified name and enters policy lookup command mode. In this case, creates a policy called “service-provider-policy”.
Step 4	sequence <i>sequence-number</i> Example: se-10-0-0-0(cusp-config-lookup)> sequence 1	Creates a sequence with the specified number and enters policy lookup sequence command mode. Sequences are performed according to the order of their number.
Step 5	rule { exact prefix subdomain subnet fixed length } [case-insensitive] Example: se-10-0-0-0(cusp-config-lookup-seq)> rule prefix	Creates a rule that determines the routing algorithm for the lookup policy. In this case, it creates a rule that specifies that the lookup policy searches for the longest prefix match.
Step 6	end sequence Example: se-10-0-0-0(cusp-config-lookup-seq)> end sequence	Exits policy lookup sequence command mode.
Step 7	end policy Example: se-10-0-0-0(cusp-config-lookup)> end policy	Exits policy lookup command mode.

Example

```

se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> policy lookup service-provider-policy
se-10-0-0-0(cusp-config-lookup)> sequence 1 service-provider-table request-uri
uri-component user
se-10-0-0-0(cusp-config-lookup-seq)> rule prefix
se-10-0-0-0(cusp-config-lookup-seq)> end sequence
se-10-0-0-0(cusp-config-lookup)> end policy

```

Configuring Routing Triggers

Routing triggers correlate trigger conditions with lookup policies. A single policy is chosen based on which corresponding condition is matched. The conditions are evaluated in ascending order based on sequence number. The mid-dialog condition is the first one so that the policy step is skipped for mid-dialog messages. Based on the following configuration, after the INVITE message is successfully routed, all subsequent messages (which are mid-dialog) bypass routing policies.

- [Summary Steps, page 11](#)
- [Detailed Steps, page 11](#)
- [Example, page 11](#)

Summary Steps

1. **culp**
2. **configure**
3. **trigger routing sequence** *sequence-number* {**by-pass** | **policy** *policy*} [**condition** *trigger-condition*]

Detailed Steps

	Command or Action	Purpose
Step 1	culp Example: se-10-0-0-0> culp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(culp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	trigger routing sequence <i>sequence-number</i> { by-pass policy <i>policy</i> } [condition <i>trigger-condition</i>] Example: se-10-0-0-0(culp-config)> trigger routing sequence 2 policy service-provider-policy condition call-from-service-provider	Associates a routing policy with a trigger condition. In this example, the second sequence follows the previously-created policy called “service-provider-policy” and the previously-created trigger called “call-from-service-provider”.

Example

```
se-10-0-0-0> culp
se-10-0-0-0(culp)> configure
se-10-0-0-0(culp-config)> trigger routing sequence 1 by-pass condition mid-dialog
se-10-0-0-0(culp-config)> trigger routing sequence 2 policy service-provider-policy
condition call-from-service-provider
```

```

se-10-0-0-0(cusp-config)> trigger routing sequence 3 policy cube-sp-policy condition
call-from-cube-sp
se-10-0-0-0(cusp-config)> trigger routing sequence 4 policy cube-es-policy condition
call-from-cube-es
se-10-0-0-0(cusp-config)> trigger routing sequence 5 policy enterprise-policy condition
call-from-enterprise

```

Configuring Normalization Triggers

Normalization triggers correlate trigger conditions with normalization policies. There are two types of triggers: pre-normalization, which occurs before routing, and post-normalization, which occurs after routing. Similar to routing policies, a special policy bypasses normalization on mid-dialog messages.

- [Summary Steps, page 12](#)
- [Detailed Steps, page 12](#)
- [Example, page 13](#)

Summary Steps

1. **cusp**
2. **configure**
3. **trigger pre-normalization sequence** *sequence-number* {**by-pass** | **policy** *policy*} [**condition** *trigger-condition*]

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	trigger pre-normalization sequence <i>sequence-number</i> { by-pass policy <i>policy</i> } [condition <i>trigger-condition</i>] Example: se-10-0-0-0(cusp-config)> trigger pre-normalization sequence 2 policy outgoing-norm-policy condition call-from-cube-sp	Configures a pre-normalization algorithm for incoming SIP messages to a normalization policy. In this example, the second sequence follows the previously-created policy called “outgoing-norm-policy” and the previously-created trigger called “call-from-cube-sp”.

Example

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> trigger pre-normalization sequence 1 by-pass condition
mid-dialog
se-10-0-0-0(cusp-config)> trigger pre-normalization sequence 2 policy outgoing-norm-policy
condition call-from-cube-sp
```

Configuring Listen and Record-Route Ports

You must configure listen and record-route ports for each network. For the listen and record-route ports, the actual addresses of the Cisco Unified SIP Proxy module are used. The **sip record-route** command inserts the record-route header into outgoing requests. The **sip listen** command allows for Cisco Unified SIP Proxy to accept incoming requests on that port.

- [Summary Steps, page 13](#)
- [Detailed Steps, page 13](#)
- [Example, page 14](#)

Summary Steps

1. **cusp**
2. **configure**
3. **sip record-route** *network_name* {**tcp** | **tls** | **udp**} *ip_address* [*port*]
4. **sip listen** *network_name* {**tcp** | **tls** | **udp**} *ip_address* *port*

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.

	Command or Action	Purpose
Step 3	sip record-route <i>network_name</i> { tcp tls udp } <i>ip_address</i> [<i>port</i>] Example: se-10-0-0-0(cusp-config) > sip record-route service-provider udp 10.10.10.99 5060	Enables record-routing for a SIP network. In this example, the “service-provider” network is associated with a record-route configuration and the IP address that populates the record-route header field is “10.10.10.99” and the port that populates the record-route header is 5060.
Step 4	sip listen <i>network_name</i> { tcp tls udp } <i>ip_address</i> <i>port</i> Example: se-10-0-0-0(cusp-config) > sip listen service-provider udp 10.10.10.99 5060	Creates a listener that listens for SIP traffic on a specific SIP network, host, and port.

Example

```

se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> sip record-route service-provider udp 10.10.10.99 5060
se-10-0-0-0(cusp-config)> sip listen service-provider udp 10.10.10.99 5060

```

Configuring a Hostname

If the upstream element is using DNS SRV for routing to the two Cisco Unified SIP Proxies in a network, you must configure the two Cisco Unified SIP Proxies to have the same FQDN by entering the **sip alias** command in Cisco Unified SIP Proxy configuration mode on both Cisco Unified SIP Proxies.

- [Summary Steps, page 14](#)
- [Detailed Steps, page 15](#)
- [Example, page 15](#)

Summary Steps

1. **cusp**
2. **configure**
3. **sip alias** *hostname*

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	sip alias hostname Example: se-10-0-0-0(cusp-config)> sip alias myhost	Configures the hostname of this instance.

Example

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> sip alias myhost
```

Configuring Transport Layer Security (TLS)

- [Creating and Importing a Signed Certificate, page 15](#)
- [Configuring TLS on Cisco Unified SIP Proxy, page 17](#)

Creating and Importing a Signed Certificate

Cisco Unified SIP Proxy supports TLS, Transmission Control Protocol (TCP), and User Datagram Protocol (UDP). Establishing TLS connections requires some extra steps because the connections require authentication using signed certificates.

- [Prerequisites, page 15](#)
- [Summary Steps, page 16](#)
- [Detailed Steps, page 16](#)
- [Example of Creating a Signed Certificate, page 16](#)

Prerequisites

You need an FTP server or HTTP to export certificate requests.

Summary Steps

1. **configure terminal**
2. **crypto key generate [rsa {label *label-name* | modulus *modulus-size*} | default]**
3. **crypto key certreq label *label-name* url {ftp: | http:}**
4. **crypto key import rsa label *label-name* {der url {ftp: | http: } | pem { terminal | url {ftp: | http: } } [default]**
5. **crypto key import cer label *mykey* url ftp:**

Detailed Steps

	Command or Action	Purpose
Step 1	configure terminal Example: se-10-0-0-0# configure terminal	Enters configuration mode.
Step 2	crypto key generate [rsa {label <i>label-name</i> modulus <i>modulus-size</i>} default] Example: se-10-0-0-0(config)> crypto key generate rsa label mykey modulus 512 default	Creates an RSA private key.
Step 3	crypto key certreq label <i>label-name</i> url {ftp: http:} Example: se-10-0-0-0(config)> crypto key certreq label mykey url ftp:	Creates a certificate request to be signed.
Step 4	crypto key import rsa label <i>label-name</i> {der url {ftp: http: } pem { terminal url {ftp: http: } } [default] Example: se-10-0-0-0(config)> crypto key import trustcacert label rootCA url ftp:	After the certificate request is signed, imports the trusted certificate authority (CA) certificate that you used to sign the request.
Step 5	crypto key import rsa label <i>label-name</i> {der url {ftp: http: } pem { terminal url {ftp: http: } } [default] Example: se-10-0-0-0(config)> crypto key import cer label mykey url ftp:	After the root CA is imported, imports the signed certificate.

Example of Creating a Signed Certificate

```
se-10-0-0-0# configure terminal
se-10-0-0-0(config)> crypto key generate rsa label mykey modulus 512 default
Key generation in progress. Please wait...
The label name for the key is mykey
```

```
se-10-0-0-0(config)> crypto key certreq label mykey url ftp:
Address or name of remote host? 192.168.202.216
Username (ENTER if none)? anonymous
Password (not shown)?
Destination path? netmod/mykey.csr
Uploading CSR file succeed

se-10-0-0-0(config)> crypto key import trustcacert label rootCA url ftp:
Import certificate file...
Address or name of remote host? 192.168.202.216
Source filename? netmod/rootCA/cacert.pem
1212 bytes received.

se-10-0-0-0(config)> crypto key import cer label mykey url ftp:
Import certificate file...
Address or name of remote host? 192.168.202.216
Source filename? netmod/mycert.cer
952 bytes received.
Import succeeded
```

What To Do Next

- Import the trusted CA certificates for any of the TLS peer elements.

Configuring TLS on Cisco Unified SIP Proxy

After you import the certificates, you must enable TLS connections. If you want more security, you can create a list of trusted peers. If you create such a list, only connections from those peers are accepted. The peer's hostname entry must be the peer's subjectAltName in its certificate. If subjectAltName is not used in the certificate, the peer's hostname entry must be CN.

- [Summary Steps, page 17](#)
- [Detailed Steps, page 18](#)
- [Example of Configuring TLS, page 18](#)

Summary Steps

1. **cusp**
2. **configure**
3. **sip tls**
4. **sip tls trusted-peer** {*peer's-hostname*}
5. **sip tls connection-setup-timeout** {*value in seconds*}

Detailed Steps

	Command or Action	Purpose
Step 1	cusp Example: se-10-0-0-0> cusp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(cusp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	sip tls Example: se-10-0-0-0(cusp-config)> sip tls	Enables the use of SIP TLS connections with other SIP entities, providing secure communication over the Internet.
Step 4	sip tls trusted-peer {peer's-hostname} Example: se-10-0-0-0(cusp-config)> sip tls trusted-peer example.com	Creates a list of trusted peers.
Step 5	sip tls connection-setup-timeout {value in seconds} Example: se-10-0-0-0(cusp-config)> sip tls connection-setup-timeout <1-60>	It is the time specified in Cisco Unified SIP Proxy by the user to establish connection with the trusted peer. The default value is 1 second. The range of values is 1 to 60 seconds.

Example of Configuring TLS

```
se-10-0-0-0> cusp
se-10-0-0-0(cusp)> configure
se-10-0-0-0(cusp-config)> sip tls
se-10-0-0-0(cusp-config)> sip tls trusted-peer example.com
se-10-0-0-0(cusp-config)> sip tls connection-setup-timeout <1-60>
```

Configuring Lite Mode

One of the ways you can configure the performance of the Cisco Unified SIP Proxy is to switch the module to Lite Mode. In Lite Mode, which requires you to disable record-route, the module's performance is boosted. In standard mode, the module processes calls up to the licensed limit.

By default, the module is in standard mode.

For information on the performance difference when using Lite Mode versus standard mode, see the [Release Notes for Cisco Unified SIP Proxy Release 9.0](#)

- [Summary Steps, page 19](#)

- [Detailed Steps, page 19](#)
- [Example, page 19](#)

Summary Steps

1. **culp**
2. **configure**
3. **lite-mode**

Detailed Steps

	Command or Action	Purpose
Step 1	culp Example: se-10-0-0-0 > culp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0 (culp) > configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	lite-mode Example: se-10-0-0-0 (culp-config) > lite-mode	Puts the Cisco Unified SIP Proxy module into Lite Mode.

Example

The following example puts the module into Lite Mode:

```
se-10-0-0-0> culp
se-10-0-0-0 (culp) > configure
se-10-0-0-0 (culp-config) > lite-mode
```

Configuring Performance Control

- [About Performance Control, page 20](#)
- [Summary Steps, page 20](#)
- [Detailed Steps, page 20](#)
- [Example, page 10](#)

About Performance Control

One of the ways you can configure the performance of the Cisco Unified SIP Proxy is to restrict the number of calls that the Cisco Unified SIP Proxy can handle.

Summary Steps

1. **culp**
2. **configure**
3. **call-rate-limit** *limit*

Detailed Steps

	Command or Action	Purpose
Step 1	culp Example: se-10-0-0-0> culp	Enters Cisco Unified SIP Proxy EXEC mode.
Step 2	configure Example: se-10-0-0-0(culp)> configure	Enters Cisco Unified SIP Proxy configuration mode.
Step 3	call-rate-limit <i>limit</i> Example: se-10-0-0-0(culp-config)> call-rate-limit 50	Sets the maximum call rate that the Cisco Unified SIP Proxy can handle.

Example

The following example limits the number of calls that the system can process to 50:

```
se-10-0-0-0> culp
se-10-0-0-0(culp)> configure
se-10-0-0-0(culp-config)> call-rate-limit 50
```

Committing the Configuration

Now you must commit the configuration. Committing the configuration serves two purposes: the configuration becomes active, and is persisted.

- To see the current active configuration, enter the **show configuration active** command.
- To see what the active configuration will be after you commit your changes, enter the **show configuration candidate** command.
- To commit the configuration for this example, enter the following command:


```
se-10-0-0-0(cusp-config) > commit
```




Backing Up and Restoring Data

Last updated: October 10, 2019



Note

Setting up a backup server is part of the initial configuration process. If you have not already done this, see [“Setting Backup Parameters” on page 3](#).

- [About Backing Up and Restoring Data, page 1](#)
- [Restrictions for Backing Up and Restoring Data, page 1](#)
- [Backing Up Files, page 2](#)
- [Restoring Files, page 4](#)
- [Related Topics, page 5](#)

About Backing Up and Restoring Data

Cisco Unified SIP Proxy backup and restore functions use an FTP server to store and retrieve data. The backup function copies the files from the Cisco Unified SIP Proxy module to the FTP server and the restore function copies the files from the FTP server to the Cisco Unified SIP Proxy application. The FTP server can reside anywhere in the network as long as the backup and restore functions can access it with an IP address or hostname.

We recommend that you back up your configuration files whenever you make changes to the system or application files. Do backups regularly to preserve configuration data.

The system supports the following types of backup:

- All—Backs up all files and data.
- Configuration—Backs up only system and application settings.
- Data—Backs up only routes and application data.

Restrictions for Backing Up and Restoring Data

- You must be in offline mode when you back up or restore the system, so we recommend performing these tasks when call traffic is least impacted. Offline mode terminates all calls.
- Cisco Unified SIP Proxy does not support the following backup and restore capabilities:

- Scheduled backup and restore operations. The backup and restore procedures begin when the appropriate command is entered.
- Centralized message storage arrangement. Cisco Unified SIP Proxy backup files cannot be used or integrated with other message stores.
- Selective backup and restore. Only full backup and restore functions are available. Individual messages or other specific data can be neither stored nor retrieved.

Backing Up Files

- [About Backing Up Files, page 2](#)
- [Summary Steps, page 2](#)
- [Detailed Steps, page 3](#)
- [Examples, page 3](#)

About Backing Up Files

Cisco Unified SIP Proxy automatically assigns a backup ID to each backup. Although there are the three different types of backups, the system does not take into account the type of backup when generating the backup ID. Therefore, you will never have two backups with the same backup ID, even if one is a configuration file and the other a data file.

To determine the backup ID of the file you want to restore, use the **show backup server** or **show backup history** commands in either EXEC or offline mode. Those commands list all available backup copies on the remote backup server and their respective backup IDs.

Summary Steps

1. **offline**
2. **backup category {all | configuration | data}**
3. **continue**
4. **show backup history**
5. **show backup server**

Detailed Steps

	Command or Action	Purpose
Step 1	offline Example: <pre>se-10-0-0-0# offline !!!WARNING!!!: Putting the system offline will terminate all active calls. Do you wish to continue[n]? : y</pre>	Enters offline mode. All calls are terminated. Note Cisco Unified SIP Proxy still routes calls in offline mode.
Step 2	backup category {all configuration data} Example: <pre>se-10-0-0-0(offline)# backup category all se-10-0-0-0(offline)# backup category configuration se-10-0-0-0(offline)# backup category data</pre>	Specifies the type of data to be backed up and stored.
Step 3	continue Example: <pre>se-10-0-0-0(offline)# continue</pre>	Exits offline mode and returns the system to the previous online mode. The system begins processing new calls and voice messages.
Step 4	show backup history Example: <pre>se-10-0-0-0> show backup history</pre>	Displays each backup file, its backup ID, the type of data stored in the file, and the success or failure of the backup procedure.
Step 5	show backup server Example: <pre>se-10-0-0-0> show backup server</pre>	Displays a list of the backup files available on the backup server. The files are grouped by category, with the date of each backup and the backup file ID.

Examples

The following examples display the output from the **show backup history** and **show backup server** commands:

```
se-10-0-0-0> show backup history

blade522> show backup history
#Start Operation
Category: Configuration
Backup Server: ftp://192.168.1.35/pub/cusp_backup
Operation: Backup
Backupid: 1
Date: Tue Oct 21 06:14:30 EDT 2008
Result: Success
Reason:
#End Operation

#Start Operation
Category: Configuration
Backup Server: ftp://192.168.1.35/pub/cusp_backup
Operation: Restore
```

```

Backupid: 1
Restoreid: 1
Date: Tue Oct 21 06:17:21 EDT 2008
Result: Success
Reason:
#End Operation

se-10-0-0-0> show backup server

Category: Data
Details of last 5 backups
Backupid: 1
Date: Tue Jul 22 10:55:52 PDT 2008
Description:
Backupid: 2
Date: Tue Jul 29 18:06:33 PDT 2008
Description:
Backupid: 3
Date: Tue Jul 29 19:10:32 PDT 2008
Description:
Category: Configuration
Details of last 5 backups
Backupid: 1
Date: Tue Jul 22 10:55:48 PDT 2008
Description:
Backupid: 2
Date: Tue Jul 29 18:06:27 PDT 2008
Description:
Backupid: 3
Date: Tue Jul 29 19:10:29 PDT 2008
Description:

se-10-0-0-0>

```

Restoring Files

- [About Restoring Files, page 4](#)
- [Summary Steps, page 4](#)
- [Detailed Steps, page 5](#)

About Restoring Files

After you create the backup files, you can restore them when needed. Restoring is done in offline mode, which terminates all calls. You should therefore consider restoring files when call traffic is least impacted.

To determine the backup ID of the file you want to restore, use the **show backup server** or **show backup history** commands in either EXEC or offline mode.

Summary Steps

1. **show backup server**
2. **offline**

3. **restore id** *backup_ID* **category** {**all** | **configuration** | **data**}
4. **show backup history**
5. **reload**

Detailed Steps

	Command or Action	Purpose
Step 1	show backup server Example: se-10-0-0-0> show backup server	Lists the data and configuration backup files. Look in the backup ID field for the revision number of the file that you want to restore.
Step 2	offline Example: se-10-0-0-0# offline !!!WARNING!!!: Putting the system offline will terminate all active calls. Do you wish to continue[n]? : y	Enters offline mode. All calls are terminated. Note Cisco Unified SIP Proxy still routes calls in offline mode.
Step 3	restore id <i>backup_ID</i> category { all configuration data } Example: se-10-0-0-0(offline)# restore id 22 category all	Specifies the backup ID value and the file type to be restored.
Step 4	show backup history Example: se-10-0-0-0> show backup history	Displays the success or failure of backup and restore procedures, and also the backup IDs.
Step 5	reload Example: se-10-0-0-0(offline)# reload	Activates the uploaded file information and restarts the Cisco Unified SIP Proxy system.

Related Topics

- For information about setting up the backup server as part of the initial configuration process, see [“Setting Backup Parameters” on page 3](#).
- For information on the CLI commands used to back up and restore the configuration, see the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#).



Maintaining the Cisco Unified SIP Proxy System

Last updated: October 10, 2019

- [Copying Configurations, page 1](#)
- [Checking Hard Disk Memory Wear Activity, page 3](#)

Copying Configurations

Use module EXEC commands to copy the startup configuration and running configuration to and from the hard disk on the Cisco Unified SIP Proxy module, the network FTP server, and the network TFTP server.



Note

Depending on the specific TFTP server you are using, you might need to create a file with the same name on the TFTP server and verify that the file has the correct permissions before transferring the running configuration to the TFTP server.

- [Copying the Startup Configuration from the Hard Disk to Another Location, page 1](#)
- [Copying the Startup Configuration from the Network FTP Server to Another Location, page 2](#)
- [Copying the Running Configuration from the Hard Disk to Another Location, page 2](#)
- [Copying the Running Configuration from the Network TFTP Server to Another Location, page 3](#)

Copying the Startup Configuration from the Hard Disk to Another Location

Starting in module EXEC mode, use the following command to copy the startup configuration on the hard disk to another location:

copy startup-config {ftp: user-id:password@ftp-server-url | tftp:tftp-server-url}

Syntax	Description
ftp: <i>user-id:password@</i>	Username and password for the FTP server. Include the colon (:) and the at sign (@) in your entry.
<i>ftp-server-url</i>	URL of the FTP server including directory and filename. An example is <code>ftp://server/dir/filename</code> .
tftp: <i>tftp-server-url</i>	URL of the TFTP server including directory and filename. An example is <code>tftp://server/dir/filename</code> .

This command is interactive and prompts you for the information. You cannot enter the parameters in one line. In this example, the startup configuration is copied to the FTP server, which requires a username and password to transfer files. The startup configuration file is saved on the FTP server with the filename “start”.

```
se-10-0-0-0> copy startup-config ftp
Address or name of remote host? admin:messaging@ftp://server/dir/start
Source filename? temp_start
```

The following example shows the startup configuration copied to the TFTP server, which does not require a username and password. The command saves the startup configuration in the TFTP directory called “configs” as a file called “temp_start”.

```
se-10-0-0-0> copy startup-config tftp
Address or name of remote host? tftp://server/dir/temp_start
Source filename? temp_start
```

Copying the Startup Configuration from the Network FTP Server to Another Location

Starting in module EXEC mode, use the following command to copy the startup configuration on the network FTP server to another location:

copy ftp: {nvram:startup-config | running-config | startup-config | system:running-config}

For a description of this command, see the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#).

This command is interactive and prompts you for the information. You cannot enter the parameters in one line. The following example illustrates this process. In this example, the FTP server requires a username and password. This command copies the file called “start” that resides in the FTP server directory called “configs” to the startup configuration.

```
se-10-0-0-0> copy ftp: startup-config
!!!WARNING!!! This operation will overwrite your startup configuration.
Do you wish to continue[y]? y
Address or name of remote host? admin:messaging@tftp://server/configs
Source filename? start
```

Copying the Running Configuration from the Hard Disk to Another Location

Starting in module EXEC mode, use the following command to copy the running configuration on the hard disk to another location:

copy running-config {ftp: user-id:password@ftp://server/dir/filename | startup-config | tftp:tftp://server/dir/filename }

For a description of this command, see the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#).

The command works in two ways, depending on where you are copying the command:

- If you copy the running configuration to the startup configuration, enter the command on one line, like in the following example:

```
se-10-0-0-0> copy running-config startup-config
```

- If you copy the running configuration to the FTP or TFTP server, this command becomes interactive and prompts you for the information. You cannot enter the parameters in one line. In the following example, the running configuration is copied to the FTP server, which requires a username and password. The running configuration is copied to the directory called “configs” as a file called “saved_start”.

```
se-10-0-0-0> copy running-config ftp:
Address or name of remote host? admin:messaging@ftps://server/configs
Source filename? saved_start
```

Copying the Running Configuration from the Network TFTP Server to Another Location

Starting in module EXEC mode, use the following command to copy the running configuration from the network TFTP server to another location:

copy tftp: {running-config | startup-config} tftp://server/dir/filename

Syntax Description

running-config	Active configuration on hard disk.
startup-config	Startup configuration on hard disk.
<i>tftp-server-url</i>	URL of the TFTP server.

This command is interactive and prompts you for the information. You cannot enter the parameters in one line. The following example illustrates this process. In this example, the file called “start” that resides in the directory called “configs” on the TFTP server is copied to the startup configuration.

```
se-10-0-0-0> copy tftp: startup-config
!!!WARNING!!! This operation will overwrite your startup configuration.
Do you wish to continue[y]? y
Address or name of remote host? tftp://server/configs
Source filename? start
```

Checking Hard Disk Memory Wear Activity

Cisco Unified SIP Proxy tracks the use and wear of the hard disk memory as log and trace data are saved to the module. To display this data, use the **show interfaces** command in module EXEC mode.

The following is sample output:

```
se-10-0-0-0> show interfaces
GigabitEthernet 0 is up, line protocol is up
  Internet address is 10.10.1.20 mask 255.255.255.0 (configured on router)
    25629 packets input, 1688582 bytes
    0 input errors, 0 dropped, 0 overrun, 0 frame errors
```

```
25634 packets output, 1785015 bytes
0 output errors, 0 dropped, 0 overrun, 0 collision errors
0 output carrier detect errors
IDE hd0 is up, line protocol is up
2060 reads, 32704512 bytes
0 read errors
489797 write, 2520530944 bytes
0 write errors
```



Troubleshooting

Last updated: October 10, 2019



Note

Use the information in this chapter in conjunction with the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#). That document contains detailed information about each CLI command listed here, including when to use it, how to use it, and any cautionary information.

This chapter contains a brief overview of troubleshooting using the CLI and contains the following sections:

- [Using CLI Commands to Troubleshoot the System, page 1](#)
- [Troubleshooting Configuration Changes, page 3](#)
- [Related Topics, page 3](#)

Using CLI Commands to Troubleshoot the System

Cisco technical support personnel may request that you run one or more of these commands when troubleshooting a problem. Cisco technical support personnel provides additional information about the commands at that time.



Caution

Some of these commands may impact performance of your system. We strongly recommend that you do not use these commands unless directed to do so by Cisco Technical Support.

- [About Logging, page 1](#)
- [Log Commands, page 2](#)
- [Example of Log Output, page 2](#)
- [Using Trace Commands, page 2](#)
- [Using Show Commands, page 3](#)

About Logging

You can use log messages to help you debug system problems. Log messages are saved to the messages.log file.

Logging and tracing to the hard disk is turned off by default. Executing the **log trace boot** command starts the log and trace functions immediately.

To check the log and trace files on the hard disk, use the **show logs** command in Cisco Unified SIP Proxy EXEC mode. It displays the list of logs available, their size and their dates of most recent modification.

Each file has a fixed length of 10 MB, and tracing or logging stops automatically when the file reaches this length. New files overwrite the old files.

**Tip**

If you cannot view the contents of the log files, copy the log files from Cisco Unified SIP Proxy to an external server and use a text editor, such as **vi**, to display the content.

Log Commands

Cisco Unified SIP Proxy has the following log commands:

- **log console** command
- **log console monitor** command
- **log server** command
- **log trace boot** command
- **log trace buffer save** command
- **show logs** command
- **show trace log** command

Example of Log Output

The following is an example of the log output:

```
se-Module(exec-mping)> show logs
```

SIZE	LAST_MODIFIED_TIME	NAME
28719	Mon Dec 22 14:15:06 EST 2008	linux_session.log
2573	Fri Dec 19 08:28:13 EST 2008	install.log
8117	Fri Dec 19 08:27:51 EST 2008	dmesg
2274	Fri Dec 19 08:27:55 EST 2008	syslog.log
10455	Thu Dec 18 16:38:13 EST 2008	sshd.log.prev
1268	Fri Dec 19 08:28:09 EST 2008	atrace.log
384	Fri Dec 19 08:27:55 EST 2008	debug_server.log
10380	Thu Dec 18 16:06:58 EST 2008	postgres.log.prev
1361	Fri Dec 19 08:28:14 EST 2008	sshd.log
5598	Fri Dec 19 08:30:13 EST 2008	postgres.log
1014	Fri Dec 19 08:27:57 EST 2008	klog.log
2298494	Sun Dec 21 23:30:00 EST 2008	messages.log
85292	Fri Dec 19 08:25:33 EST 2008	shutdown_installer.log

Using Trace Commands

To troubleshoot network configuration in Cisco Unified SIP Proxy, use the **trace enable** command in Cisco Unified SIP Proxy EXEC mode.

Cisco Unified SIP Proxy has the following trace commands:

- **log trace boot** command
- **log trace buffer save** command
- **show trace log** command
- **show trace options** command
- **trace disable** command
- **trace enable** command
- **trace level** command

Using Show Commands

In addition to the standard show commands, use the following commands to troubleshoot your Cisco Unified SIP Proxy configuration:

- **show status queue**
- **show status server-group radius** [*server-group-name*]
- **show status server-group sip** [*server-group-name*]
- **show status sip**

Troubleshooting Configuration Changes

Problem You lost some configuration data.

Recommended Action Copy your changes to the running configuration at frequent intervals. See [“Copying Configurations” on page 1](#).

Problem You lost configuration data when you rebooted the system.

Explanation You did not save the data before the reboot.

Recommended Action Use the **copy running-config startup-config** command to copy your changes from the running configuration to the startup configuration. When Cisco Unified SIP Proxy reboots, it reloads the startup configuration. See [“Copying Configurations” on page 1](#).



Note

Messages are considered application data and are saved directly to the disk in the startup configuration. (They should be backed up on another server in case of a power outage or a new installation.) All other configuration changes require an explicit “save configuration” operation to preserve them in the startup configuration.

Related Topics

- For information about the CLI commands, see the [CLI Command Reference for Cisco Unified SIP Proxy Release 9.0](#).
- For information about copying configurations, see [“Copying Configurations” on page 1](#).



Configuration Example

Last updated: October 10, 2019

The following is an example of what you will see after you have configured your Cisco Unified SIP Proxy system and then enter the **show configuration active verbose** command.

```
se-10-0-0-0(cusp-config)> show configuration active verbose
Building CUSP configuration...
!
server-group sip global-load-balance call-id
server-group sip retry-after 0
server-group sip element-retries udp 3
server-group sip element-retries tls 1
server-group sip element-retries tcp 1
sip alias myhostname
sip dns-srv
    enable
    no naptr
end dns
!
no sip header-compaction
no sip logging
!
sip max-forwards 70
sip network cube-es standard
    no non-invite-provisional
    allow-connections
    retransmit-count invite-server-transaction 9
    retransmit-count non-invite-client-transaction 9
    retransmit-count invite-client-transaction 5
    retransmit-timer clientTn 64000
    retransmit-timer serverTn 64000
    retransmit-timer T4 5000
    retransmit-timer T2 4000
    retransmit-timer T1 500
    retransmit-timer TU2 32000
    retransmit-timer TU1 5000
end network
!
sip network cube-sp standard
    no non-invite-provisional
    allow-connections
    retransmit-count invite-server-transaction 9
    retransmit-count invite-client-transaction 5
    retransmit-count non-invite-client-transaction 9
    retransmit-timer T4 5000
    retransmit-timer T2 4000
    retransmit-timer T1 500
```

```

retransmit-timer TU2 32000
retransmit-timer TU1 5000
retransmit-timer clientTn 64000
retransmit-timer serverTn 64000
end network
!
sip network enterprise standard
no non-invite-provisional
allow-connections
retransmit-count invite-client-transaction 5
retransmit-count invite-server-transaction 9
retransmit-count non-invite-client-transaction 9
retransmit-timer serverTn 64000
retransmit-timer T4 5000
retransmit-timer T2 4000
retransmit-timer T1 500
retransmit-timer TU2 32000
retransmit-timer TU1 5000
retransmit-timer clientTn 64000
end network
!
sip network service-provider standard
no non-invite-provisional
allow-connections
retransmit-count invite-server-transaction 9
retransmit-count non-invite-client-transaction 9
retransmit-count invite-client-transaction 5
retransmit-timer serverTn 64000
retransmit-timer TU1 5000
retransmit-timer TU2 32000
retransmit-timer T1 500
retransmit-timer T2 4000
retransmit-timer T4 5000
retransmit-timer clientTn 64000
end network
!
sip overload reject retry-after 0
!
no sip peg-counting
!
sip privacy service
sip queue message
drop-policy head
low-threshold 80
size 2000
thread-count 20
end queue
!
sip queue radius
drop-policy head
low-threshold 80
size 2000
thread-count 20
end queue
!
sip queue request
drop-policy head
low-threshold 80
size 2000
thread-count 20
end queue
!
sip queue response
drop-policy head

```

```

low-threshold 80
size 2000
thread-count 20
end queue
!
sip queue st-callback
drop-policy head
low-threshold 80
size 2000
thread-count 10
end queue
!
sip queue timer
drop-policy none
low-threshold 80
size 2500
thread-count 8
end queue
!
sip queue xcl
drop-policy head
low-threshold 80
size 2000
thread-count 2
end queue
!
route recursion
!
sip tcp connection-timeout 240
sip tcp max-connections 256
sip tls
!
trigger condition call-from-cube-es
sequence 1
in-network cube-es
end sequence
end trigger condition
!
trigger condition call-from-cube-sp
sequence 1
in-network cube-sp
end sequence
end trigger condition
!
trigger condition call-from-enterprise
sequence 1
in-network enterprise
end sequence
end trigger condition
!
trigger condition call-from-service-provider
sequence 1
in-network service-provider
end sequence
end trigger condition
!
trigger condition mid-dialog
sequence 1
mid-dialog
end sequence
end trigger condition
!
accounting
no enable

```

```

no client-side
no server-side
end accounting
!
server-group sip group cme.example.com enterprise
element ip-address 192.168.10.6 5060 tls q-value 1.0 weight 0
failover-resp-codes 503
lbtype global
ping
end server-group
!
server-group sip group cube-es.example.com cube-es
element ip-address 192.168.20.4 5060 tls q-value 1.0 weight 0
element ip-address 192.168.20.3 5060 tls q-value 1.0 weight 0
failover-resp-codes 503
lbtype global
ping
end server-group
!
server-group sip group cube-sp.example.com cube-sp
element ip-address 10.10.20.3 5060 tls q-value 1.0 weight 0
element ip-address 10.10.20.4 5060 tls q-value 1.0 weight 0
failover-resp-codes 503
lbtype global
ping
end server-group
!
server-group sip group cucm.example.com enterprise
element ip-address 192.168.10.4 5060 tls q-value 1.0 weight 50
element ip-address 192.168.10.5 5060 tls q-value 1.0 weight 50
element ip-address 192.168.10.3 5060 tls q-value 1.0 weight 100
failover-resp-codes 503
lbtype weight
ping
end server-group
!
server-group sip group sp.example.com service-provider
element ip-address 10.10.10.3 5060 udp q-value 1.0 weight 0
failover-resp-codes 503
lbtype global
ping
end server-group
!
route table cube-es-table
key * response 404
key 5101 target-destination cme.example.com enterprise
key 510 target-destination cucm.example.com enterprise
end route table
!
route table cube-sp-table
key * target-destination sp.example.com service-provider
end route table
!
route table enterprise-table
key * response 404
key 5101 target-destination cme.example.com enterprise
key 91 target-destination cube-es.example.com cube-es
key 510 target-destination cucm.example.com enterprise
end route table
!
route table service-provider-table
key * response 404
key 510 target-destination cube-sp.example.com cube-sp
end route table

```

```

!
policy normalization outgoing-norm-policy
  uri-component update TO all user ^91 ""
  uri-component update request-uri user ^91 ""
end policy
!
policy lookup cube-es-policy
  sequence 1 cube-es-table request-uri uri-component user
  rule prefix
  end sequence
end policy
!
policy lookup cube-sp-policy
  sequence 1 cube-sp-table request-uri uri-component user
  rule prefix
  end sequence
end policy
!
policy lookup enterprise-policy
  sequence 1 enterprise-table request-uri uri-component user
  rule prefix
  end sequence
end policy
!
policy lookup service-provider-policy
  sequence 1 service-provider-table request-uri uri-component user
  rule prefix
  end sequence
end policy
!
trigger routing sequence 5 policy enterprise-policy condition call-from-enterpri
se
trigger routing sequence 4 policy cube-es-policy condition call-from-cube-es
trigger routing sequence 3 policy cube-sp-policy condition call-from-cube-sp
trigger routing sequence 2 policy service-provider-policy condition
call-from-service-provider
trigger routing sequence 1 by-pass condition mid-dialog
trigger pre-normalization sequence 2 policy outgoing-norm-policy condition
call-from-cube-sp
trigger pre-normalization sequence 1 by-pass condition mid-dialog
!
no server-group sip global-ping
!
sip listen service-provider udp 10.10.10.99 5060
sip listen cube-sp tls 10.10.20.99 5060
sip listen cube-es tls 192.168.20.99 5060
sip listen enterprise tls 192.168.10.99 5060
!
sip record-route cube-es tls 192.168.20.99 5060
sip record-route service-provider udp 10.10.10.99 5060
sip record-route cube-sp tls 10.10.20.99 5060
sip record-route enterprise tls 192.168.10.99 5060
!
end
se-10-0-0-0(cusp-config)>

```




B

backup

- configuring [7](#)
- FTP server [7, 35](#)
- parameters [7](#)
- restrictions [35](#)

backup category command [36](#)

backup revisions number command [8](#)

backup server url command [8](#)

C

call-rate-limit command [34](#)

certificate, creating [29](#)

Cisco Unified Communications Manager [18](#)

clock timezone command [13](#)

command

- backup category [36](#)
- backup revisions number [8](#)
- backup server url [8](#)
- call-rate-limit [34](#)
- clock timezone [13](#)
- configure [16, 17, 19, 20, 22, 23, 25, 26, 27, 28, 31, 32, 33](#)
- configure terminal [8, 10, 11, 13, 30](#)
- continue [36](#)
- copy ftp [42](#)
- copy running-config [42](#)
- copy running-config startup-config [10, 11, 47](#)
- copy startup-config [41](#)
- copy tftp [43](#)
- crypto key certreq label [30](#)
- crypto key generate [30](#)

crypto key import cer label [30](#)

crypto key import rsa label [30](#)

culp [15, 17, 19, 20, 22, 23, 25, 26, 27, 28, 31, 32, 33](#)

element ip-address [19](#)

end [8, 10](#)

end network [16](#)

end policy [22, 23](#)

end route table [21](#)

end sequence [17, 23](#)

end server-group [19](#)

end trigger condition [17](#)

exit [11](#)

in-network [17](#)

key response [20](#)

key target-destination [20](#)

lb-type [19](#)

lite-mode [33](#)

log console [46](#)

log console monitor [46](#)

log server [46](#)

log trace boot [46, 47](#)

log trace buffer save [46, 47](#)

mid-dialog [17](#)

ntp server [10, 11](#)

offline [36, 38](#)

policy lookup [23](#)

policy normalization [22](#)

reload [39](#)

restore id [39](#)

route table [20](#)

rule [23](#)

sequence [17, 23](#)

server-group sip group [19](#)

- show backup 8
 - show backup history 36, 39
 - show backup server 36, 38
 - show clock detail 13
 - show configuration active 34
 - show configuration candidate 34
 - show interfaces 43
 - show logs 46
 - show ntp associations 12
 - show ntp configuration 10, 11
 - show ntp servers 12
 - show ntp source 12
 - show ntp status 10, 11, 12
 - show status queue 47
 - show status server-group radius 47
 - show status server-group sip 47
 - show status sip 47
 - show trace log 46, 47
 - show trace options 47
 - sip alias 28
 - sip listen 27
 - sip network 16
 - sip record-route 27
 - sip tls 31
 - sip tls trusted-peer 31
 - trace disable 47
 - trace enable 47
 - trace level 47
 - trigger condition 17
 - trigger pre-normalization sequence 26
 - trigger routing sequence 25
 - uri-component update header 22
 - uri-component update request-uri 22
 - command-line interface
 - about 1
 - committing the configuration 34
 - configuration
 - committing 34
 - copying 41
 - configuration tasks
 - configuring hostname 28
 - configuring listen and record-route ports 27
 - configuring logical networks 15
 - configuring lookup policies 23
 - configuring normalization policies 22
 - configuring normalization triggers 26
 - configuring NTP servers 9
 - configuring route tables 20
 - configuring routing triggers 25
 - configuring server groups 18
 - configuring TLS 29
 - configuring trigger conditions 16
 - configure command 16, 17, 19, 20, 22, 23, 25, 26, 27, 28, 31, 32, 33
 - configure terminal command 8, 10, 11, 13, 30
 - continue command 36
 - copy ftp command 42
 - copying
 - configurations 41
 - copy running-config command 42
 - copy running-config startup-config command 10, 11, 47
 - copy startup-config command 41
 - copy tftp command 43
 - crypto key certreq label command 30
 - crypto key generate comamnd 30
 - crypto key import cer label command 30
 - crypto key import rsa label command 30
 - cusp command 15, 17, 19, 20, 22, 23, 25, 26, 27, 28, 31, 32, 33
-
- ## D
- displaying
 - NTP server 12
 - DNS server
 - resolving host name to IP address 9

E

element ip-address command [19](#)
 end command [8, 10](#)
 end network command [16](#)
 end policy command [22, 23](#)
 end route table command [21](#)
 end sequence command [17, 23](#)
 end server-group command [19](#)
 end trigger condition command [17](#)
 exit command [11](#)

F

file size
 messages.log [46](#)
 FTP server
 backup and restore [7, 35](#)
 copying startup configuration from [42](#)
 Fully Qualified Domain Name, configuring [18](#)

G

graphical user interface
 about [1, 2](#)

H

hard disk
 copying configuration from [42](#)
 copying startup configuration from [41](#)
 logs [46](#)
 wear [43](#)
 hostnames, about [28](#)
 hostnames, configuring [28](#)

I

initial configuration tasks
 configuring backup parameters [7](#)
 configuring NTP servers [9](#)
 setting the time zone [13](#)
 in-network command [17](#)

K

key response command [20](#)
 key target-destination command [20](#)

L

lb-type command [19](#)
 licenses
 installing [3](#)
 listen and record-route ports, configuring [27](#)
 lite-mode command [33](#)
 log console command [46](#)
 log console monitor command [46](#)
 logging, about [45](#)
 log messages [45](#)
 log server command [46](#)
 log trace boot command [46, 47](#)
 log trace buffer save command [46, 47](#)
 lookup policies
 about [23](#)
 configuring [23](#)
 lost data, troubleshooting [47](#)

M

messages.log, file size [46](#)
 mid-dialog command [17](#)
 module
 usage [43](#)

wear [43](#)

N

normalization policies

about [22](#)

configuring [22](#)

normalization triggers, about [26](#)

normalization triggers, configuring [26](#)

NTP server

configuring [9](#)

displaying [12](#)

removing [11](#)

ntp server command [10, 11](#)

O

offline command [36, 38](#)

offline mode [38](#)

P

parameters

backup [7](#)

policy lookup command [23](#)

policy normalization command [22](#)

Q

q-value [18](#)

R

record-route

about ports [27](#)

configuring ports [27](#)

reload command [39](#)

removing an NTP server [11](#)

resolving host name to IP address [9](#)

restore

FTP server [7, 35](#)

procedure [38](#)

restrictions [35](#)

restore id command [39](#)

restrictions

backup and restore [35](#)

route table command [20](#)

route tables

about [20](#)

configuring [20](#)

routing triggers

about [25](#)

configuring [25](#)

rule command [23](#)

S

sequence command [17, 23](#)

server groups

about [18](#)

configuring [18](#)

server-group sip group command [19](#)

show backup command [8](#)

show backup history command [36, 39](#)

show backup server command [36, 38](#)

show clock detail command [13](#)

show commands [47](#)

show configuration active command [34](#)

show configuration candidate command [34](#)

show interfaces command [43](#)

show logs command [46](#)

show ntp associations command [12](#)

show ntp configuration command [10, 11](#)

show ntp servers command [12](#)

show ntp source command [12](#)

show ntp status command [10, 11, 12](#)

show status queue command [47](#)

show status server-group radius command 47
show status server-group sip command 47
show status sip command 47
show trace log command 46, 47
show trace options command 47
sip alias command 28
sip listen command 27
sip network command 16
sip record-route command 27
sip tls command 31
sip tls trusted-peer command 31

T

TFTP server, copying configuration from 43
time zone 13
TLS, configuring 29
trace commands 46
trace disable command 47
trace enable command 47
trace level command 47
Transmission Control Protocol 29
transport layer security, configuring 29
trigger condition command 17
trigger conditions
 configuring 16
trigger pre-normalization sequence command 26
trigger routing sequence command 25
troubleshooting 45
 lost data 47
 using show commands 47
 using trace commands 46

U

uri-component update header command 22
uri-component update request-uri command 22
User Datagram Protocol 29

