

Command Line Interface

Root-Level Commands

Version 7.2

Table of Contents

| | | |
|---------------------------------|--|-----------|
| 1 | Introduction..... | 9 |
| 1.1 | Typographical Conventions | 10 |
| <hr/> | | |
| Getting Started..... | | 11 |
| 2 | Accessing the CLI | 13 |
| 3 | CLI Structure..... | 15 |
| 3.1 | CLI Command Modes | 15 |
| 3.1.1 | Basic Mode | 15 |
| 3.1.2 | Enable Mode..... | 15 |
| 3.1.3 | Switching between Command Modes..... | 16 |
| 3.2 | CLI Configuration Wizard..... | 16 |
| 3.3 | CLI Shortcut Keys..... | 17 |
| 3.4 | Common CLI Commands | 18 |
| 3.5 | Working with Tables | 19 |
| 3.5.1 | Adding New Rows..... | 19 |
| 3.5.2 | Adding New Rows to Specific Indices..... | 20 |
| 3.5.3 | Changing Index Position of Rows | 21 |
| 3.6 | CLI Error Messages..... | 22 |
| <hr/> | | |
| Root-Level Commands..... | | 23 |
| 4 | Introduction..... | 25 |
| 5 | Debug Commands..... | 27 |
| 5.1 | debug auxiliary-files..... | 27 |
| 5.1.1 | debug auxiliary-files dial-plan | 28 |
| 5.1.2 | debug auxiliary-files user-info | 29 |
| 5.2 | debug capture..... | 29 |
| 5.2.1 | debug capture trim | 30 |
| 5.2.2 | debug capture voip..... | 30 |
| 5.2.2.1 | debug capture voip interface | 31 |
| 5.2.2.2 | debug capture voip physical | 33 |
| 5.3 | debug debug-recording..... | 35 |
| 5.4 | debug dial plan | 36 |
| 5.5 | debug fax..... | 37 |
| 5.6 | debug log..... | 38 |
| 5.7 | debug pstn..... | 39 |
| 5.8 | debug sip..... | 40 |
| 5.9 | debug speedtest | 41 |
| 5.10 | debug syslog | 42 |
| 5.11 | debug syslog-server | 43 |
| 5.12 | debug test-call | 44 |
| 5.13 | debug usb..... | 45 |
| 5.14 | debug voip..... | 46 |

| | | |
|----------|-----------------------------------|-----------|
| 6 | Show Commands | 47 |
| 6.1 | show active-alarms | 47 |
| 6.2 | show alarms-history | 48 |
| 6.3 | show clock | 48 |
| 6.4 | show high-availability | 49 |
| 6.5 | show network | 50 |
| 6.6 | show ntp | 51 |
| 6.7 | show radius | 52 |
| 6.8 | show running-config | 53 |
| 6.9 | show storage-history | 54 |
| 6.10 | show system | 55 |
| 6.11 | show tls | 56 |
| 6.12 | show users | 57 |
| 6.13 | show version | 57 |
| 6.14 | show voip | 58 |
| 6.14.1 | show voip calls | 58 |
| 6.14.1.1 | show voip calls active | 59 |
| 6.14.1.2 | show voip calls history | 60 |
| 6.14.1.3 | show voip calls statistics | 61 |
| 6.14.2 | show voip channel-stats | 62 |
| 6.14.3 | show voip coders-stats | 63 |
| 6.14.4 | show voip cpu-stats | 63 |
| 6.14.5 | show voip dsp | 64 |
| 6.14.5.1 | show voip dsp perf | 64 |
| 6.14.5.2 | show voip dsp status | 64 |
| 6.14.6 | show voip e911 | 65 |
| 6.14.7 | show voip ids | 66 |
| 6.14.8 | show voip interface | 67 |
| 6.14.9 | show voip ip-group | 69 |
| 6.14.10 | show voip ldap | 70 |
| 6.14.11 | show voip other-dialog | 71 |
| 6.14.12 | show voip proxy | 71 |
| 6.14.13 | show voip realm | 72 |
| 6.14.14 | show voip register | 72 |
| 6.14.15 | show voip subscribe | 74 |
| 7 | Clear Commands | 75 |
| 7.1 | clear alarms-history | 75 |
| 7.2 | clear debug-file | 75 |
| 7.3 | clear qos | 75 |
| 7.4 | clear storage-history | 76 |
| 7.5 | clear system-log | 76 |
| 7.6 | clear user | 77 |
| 7.7 | clear voip | 78 |
| 7.7.1 | clear voip calls | 78 |
| 7.7.2 | clear voip register | 79 |
| 7.7.3 | clear voip statistics | 80 |
| 8 | Maintenance Commands | 81 |
| 8.1 | admin register | 82 |
| 8.2 | copy | 83 |
| 8.3 | dir | 86 |

| | | |
|------|--------------------------------------|----|
| 8.4 | erase | 87 |
| 8.5 | nslookup | 88 |
| 8.6 | ping..... | 89 |
| 8.7 | pstn nfas-group-switch-activity..... | 90 |
| 8.8 | reload | 91 |
| 8.9 | srd-view | 92 |
| 8.10 | telnet..... | 93 |
| 8.11 | traceroute | 94 |
| 8.12 | usb..... | 95 |
| 8.13 | write..... | 95 |
| 8.14 | write factory | 96 |
| 8.15 | write-and-backup | 97 |

List of Tables

| | |
|--|----|
| Table 1-1: Typographical Conventions | 10 |
| Table 3-1: CLI Shortcut Keys..... | 17 |
| Table 3-2: Common CLI Commands..... | 18 |
| Table 3-3: CLI Error Messages..... | 22 |

Notice

This document describes the root-level Command Line Interface (CLI) commands for the AudioCodes Mediant product line.

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Updates to this document and other documents can be viewed by registered customers at <http://www.audiocodes.com/downloads>.

© Copyright 2016 AudioCodes Ltd. All rights reserved.

This document is subject to change without notice.

Date Published: Sep-18-2016

Trademarks

AudioCodes, AC, HD VoIP, HD VoIP Sounds Better, IPmedia, Mediant, MediaPack, What's Inside Matters, OSN, SmartTAP, VMAS, VoIPerfect, VoIPerfectHD, Your Gateway To VoIP, 3GX, VocaNom, AudioCodes One Voice and CloudBond are trademarks or registered trademarks of AudioCodes Limited. All other products or trademarks are property of their respective owners. Product specifications are subject to change without notice.

WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

Customer Support

Customer technical support and services are provided by AudioCodes or by an authorized AudioCodes Service Partner. For more information on how to buy technical support for AudioCodes products and for contact information, please visit our Web site at www.audiocodes.com/support.

Abbreviations and Terminology

Each abbreviation, unless widely used, is spelled out in full when first used.

Document Revision Record

| LTRT | Description |
|-------|---|
| 17922 | Initial document release for Version 7.2. |

Related Documentation

| Manual Name |
|---|
| Mediant 500 E-SBC User's Manual |
| Mediant 500L E-SBC User's Manual |
| Mediant 800B Gateway & E-SBC User's Manual |
| Mediant 1000B Gateway & E-SBC User's Manual |
| Mediant 2600 E-SBC User's Manual |
| Mediant 4000 SBC User's Manual |
| Mediant 9000 SBC User's Manual |
| Mediant Software SBC User's Manual |

Documentation Feedback

AudioCodes continually strives to produce high quality documentation. If you have any comments (suggestions or errors) regarding this document, please fill out the Documentation Feedback form on our Web site at <http://www.audiocodes.com/downloads>.

1 Introduction

This document describes the root-level Command-Line Interface (CLI) commands for AudioCodes devices.

**Note:**

- For commands related to configuration, refer to the *User's Manual*.
- Some AudioCodes products referred to in this document may not have been released in Version 7.2 so commands that apply only to them should be ignored. For a list of the AudioCodes products released in Version 7.2, refer to the *Release Notes Ver. 7.2*.

1.1 Typographical Conventions

This document uses the following typographical conventions:

Table 1-1: Typographical Conventions

| Convention | Description |
|------------------|---|
| Bold font | Indicates commands and subcommands. # ping 10.4.0.1 timeout 10 |
| < > | Indicates a value (digits or characters) that you need to enter, for example: # ping < IP address > timeout < Duration > |
| | Indicates a choice between commands or sub-commands, for example: # reload { if-needed now without-saving } |
| [] | Indicates optional commands (not mandatory). The following example shows two optional commands, size and repeat : # ping < IP Address > timeout < Duration > [size < Max Packet Size >] [repeat < 1-300 >] |
| { } | Indicates a choice between commands where it is mandatory to select one of them, for example: # reload { if-needed now without-saving } |



Part I

Getting Started

2 Accessing the CLI

You can access the device's CLI through the following:

- **RS-232:** The device can be accessed through RS-232 by connecting a VT100 terminal to the device or using a terminal emulation program (e.g., HyperTerminal®) with a PC. Once you have connected via a VT100 terminal and started the emulation program, set the program settings as follows:
 - 115200 baud rate
 - 8 data bits
 - No parity
 - 1 stop bit
 - No flow control

For information on cabling your device's RS-232 interface, refer to the *User's Manual* or *Hardware Installation Manual*.

- **SSH:** The device can be accessed by SSH protocol using SSH client software. A popular freeware SSH client software is PuTTY, which can be downloaded from <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>. By default, SSH access is disabled. To enable SSH, enter the following command set:

```
# configure system
(config-system)# cli-settings
(cli-settings)# ssh on
```

- **Telnet:** The device can be accessed by Telnet protocol using Telnet client software. Most Windows® computers come with a program called Telnet, which can be activated via the Windows command-line:

```
> telnet < Device's OAMP IP Address >
Welcome to ...
Username: < Username >
Password: < Password >
```



Note:

- The CLI can only be accessed by management users with the following user levels:
 - ✓ Administrator
 - ✓ Security Administrator
 - ✓ Master
- When access the device's CLI, you are prompted to enter your management username and password. The credentials are common to all the device's management interfaces (e.g. Web). The default Administrator's username and password is "Admin" and "Admin", respectively.

This page is intentionally left blank.

3 CLI Structure

This section describes the CLI structure.

3.1 CLI Command Modes

Before you begin your CLI session, you should familiarize yourself with the CLI command modes. Each mode provides different levels of access to commands, as described below.

3.1.1 Basic Mode

The Basic command mode is accessed upon a successful CLI login authentication. Any user level can access the mode. The commands available under this mode are limited and only allow you to view information (using the `show` commands) and activate various debugging capabilities.

```
Welcome to ...
Username: Admin
Password: < Password >
>
```

The Basic mode prompt is ">".

3.1.2 Enable Mode

The Enable command mode is the high-level tier in the command hierarchy, one step up from the Basic mode. A password is required to access the mode **after** you have accessed the Basic mode. The mode allows you to configure all the device's settings. Once you have logged in to the device, the Enable mode is accessed by entering the following commands:

```
> enable
Password: < Enable mode password >
#
```

The Enable mode prompt is "#".



Note: The default password for accessing the Enable mode is "Admin" (case-sensitive). To change this password, use the `CLIPrivPass` ini file parameter.

The Enable mode groups the configuration commands under the following command sets:

- **Network:** Contains IP network-related commands (e.g., interface and dhcp-server):

```
# configure network
(config-network)#
```

- **System:** Contains system-related commands (e.g., clock, snmp settings, and web):

```
# configure system
(config-system)#
```

- **Troubleshoot:** Contains logging-related commands (e.g., syslog, logging and test-call):

```
# configure troubleshoot
(config-troubleshoot)#
```

- **Voip:** Contains voice-over-IP related commands (e.g., ip-group, sbc, gateway, and media):

```
# configure voip
(config-voip)#
```

3.1.3 Switching between Command Modes

To switch between command modes, use the following commands on the root-level prompt:

- Switching from Disable to Enable mode:

```
> enable
Password: < Password >
#
```

- Switching from Enable to Disable mode:

```
# disable
>
```

3.2 CLI Configuration Wizard

AudioCodes CLI Wizard provides a quick-and-easy tool for configuring your device with basic, initial management settings:

- Login passwords of the Security Administrator ("Admin") and User Monitor user accounts for accessing the device's embedded Web and CLI servers.
- IP network of the operations, administration, maintenance, and provisioning (OAMP) interface
- SNMP community strings (read-only and read-write)

The utility is typically used for first-time configuration of the device and is performed through a direct RS-232 serial cable connection with a computer. Configuration is done using the device's CLI. Once configured through the utility, you can access the device's management interface through the IP network.

To access the CLI Wizard, enter the following command at the root-prompt level:

```
# configure-wizard
```

For more information on how to use this utility, refer to the *CLI Wizard User's Guide*.

3.3 CLI Shortcut Keys

The device's CLI supports the following shortcut keys to facilitate configuration.

Table 3-1: CLI Shortcut Keys

| Shortcut Key | Description |
|--------------|---|
| ↑ | (Up arrow key) Retypes the previously entered command. Continuing to press the key cycles through all commands entered, starting with the most recent command. |
| Tab | Pressing the key after entering a partial, but unique command automatically completes the command name. |
| ? | <p>(Question mark) Can be used for the following:</p> <ul style="list-style-type: none"> To display commands pertaining to the command set, for example: <pre>(config-network)#? access-list Network access list dhcp-server DHCP server configuration dns DNS configuration ...</pre> To display commands beginning with certain letters. Enter the letter followed by the "?" mark, for example: <pre>(config-network)#d? dhcp-server DHCP server configuration dns DNS configuration</pre> To display a description of a command. Enter the command followed by the "?" mark (no spaces), for example: <pre>(config-network)#dns srv2ip? srv2ip SRV to IP internal table</pre> To display all subcommands for the current command. Enter the command, a space, and then the "?" mark, for example: <pre>(config-network)# dns srv2ip ? [0-9] index ...</pre> <p>If one of the listed items after running the "?" mark is "<cr>", a carriage return (Enter) can be entered to run the command, for example:</p> <pre># show active-alarms ? <cr></pre> |
| Ctrl + A | Moves the cursor to the beginning of the command line. |
| Ctrl + E | Moves the cursor to the end of the command line. |
| Ctrl + U | Deletes all characters on the command line. |
| Space Bar | When pressed after "--MORE--" that appears at the end of a displayed list, the next items are displayed. |

3.4 Common CLI Commands

The table below describes common CLI commands.

Table 3-2: Common CLI Commands

| Command | Description |
|-------------------------|---|
| do | Runs a command from another unrelated command without exiting the current command set. For example, the command to display all active alarms is run from the current command set for clock settings: <pre>(clock)# do show active-alarms</pre> |
| no | Undoes an issued command or disables a feature. Enter the no before the command: <pre># no debug log</pre> |
| activate | Activates a command. Commands are not applied until you enter the activate or exit commands. Note: Offline configuration changes require a reset of the device. A reset can be performed at the end of the configuration changes. A required reset is indicated by an asterisk (*) before the command prompt. |
| exit | Leaves the current command-set and returns one level up. For online parameters, if the configuration was changed and no activate command was entered, the exit command applies the activate command automatically. If entered on the top level, the session ends. <pre>(config-system)# exit # exit Connection to host lost.</pre> |
| display | Displays the configuration of current configuration set. |
| help | Displays a short help how-to string. |
| history | Displays a list of previously run commands. |
| list | Displays the available command list of the current command-set. |
| <filter> | Applied to a command output. The filter should be typed after the command with a pipe mark (). Supported filters: <ul style="list-style-type: none"> ▪ include <word> – filter (print) lines which contain <word> ▪ exclude <word> – filter lines which does not contain <word> ▪ grep <options> - filter lines according to <i>grep</i> common Unix utility options ▪ egrep <options> - filter lines according to <i>egrep</i> common Unix utility options ▪ begin <word> – filter (print) lines which begins with <word> ▪ between <word1> <word2> – filter (print) lines which are placed between <word1> and <word2> ▪ count – show the output's line count Example: <pre># show system version grep Number ;Serial Number: 2239835;Slot Number: 1</pre> |

3.5 Working with Tables

This section describes general commands for configuring tables in the CLI.

3.5.1 Adding New Rows

When you add a new row to a table, it is automatically assigned to the next consecutive, available index.

Syntax

```
# < table name > new
```

Command Mode

Enable

Example

If the Accounts table is configured with three existing rows (account-0, account-1, and account-2) and a new row is added, account-3 is automatically created and its configuration mode is accessed:

```
(config-voip)# sip-definition account new  
(account-3)#
```

3.5.2 Adding New Rows to Specific Indices

You can add a new row to any specific index number in the table, even if a row has already been configured for that index. The row that was assigned that index is incremented to the next consecutive index number, as well as all the index rows listed below it in the table.

Syntax

```
# < table name > < row index > insert
```

Note

The command is applicable only to the following tables:

- **SBC:**
 - IP-to-IP Routing
 - Classification
 - Message Condition
 - IP-to-IP Inbound Manipulation
 - IP-to-IP Outbound Manipulation
- **SBC and Gateway:**
 - Message Manipulations
- **Gateway:**
 - Destination Phone Number Manipulation Tables for IP-to-Tel / Tel-to-IP Calls
 - Calling Name Manipulation Tables for IP-to-Tel / Tel-to-IP Calls
 - Source Phone Number Manipulation Tables IP-to-Tel / Tel-to-IP Calls
 - Redirect Number Tel-to-IP

Command Mode

Enable

Example

If the IP-to-IP Routing table is configured with three existing rows (ip2ip-routing-0, ip2ip-routing-1, and ip2ip-routing-2) and a new row is added at Index 1, the previous ip2ip-routing-1 becomes ip2ip-routing-2, the previous ip2ip-routing-2 becomes ip2ip-routing-3, and so on:

```
(config-voip)# sbc routing ip2ip routing 1 insert
(ip2ip-routing-1)#
```

3.5.3 Changing Index Position of Rows

You can change the position (index) of a table row, by moving it one row up or one row down in the table.

Syntax

```
# < table name > < row index > move-up | move-down
```

Note

The command is applicable only to certain tables.

Command Mode

Enable

Example

Moving row at Index 1 down to Index 2 in the IP-to-IP Routing table:

```
<config-voip># sbc routing ip2ip-routing 1 move-down
```

3.6 CLI Error Messages

The table below lists and defines common error messages given in the CLI.

Table 3-3: CLI Error Messages

| Message | Helpful Hints |
|----------------------|--|
| "Invalid command" | The command may be invalid in the current command mode, or you may not have entered enough correct characters for the command to be recognized. Try using "?" to determine your error. |
| "Incomplete command" | You may not have entered all of the pertinent information required to make the command valid. Try using "?" to determine your error. |



Note:

- Optional arguments in commands are marked in square brackets [].
- To ensure that all failed CLI commands' error/information messages are displayed in the CLI console, you can redirect these messages, received from the Syslog console, to the CLI console by running the `debug log` command.



Part II

Root-Level Commands

4 Introduction

This part describes the commands located at the root level:

- Debug commands (see Section 5)
- Show commands (see Section 6)
- Clear commands (see Section 7)
- Maintenance commands (see Section 8)

This page is intentionally left blank.

5 Debug Commands

This chapter describes the following debug commands:

- `debug auxiliary-files` (see Section 5.1)
- `debug capture` (see Section 5.2)
- `debug debug-recording` (see Section 5.3)
- `debug dial-plan` (see Section 5.4)
- `debug fax` (see Section 5.5)
- `debug log` (see Section 5.6)
- `debug pstn` (see Section 5.7)
- `debug sip` (see Section 5.8)
- `debug speedtest` (see Section 5.9)
- `debug syslog` (see Section 5.10)
- `debug syslog-server` (see Section 5.11)
- `debug test-call` (see Section 5.12)
- `debug usb` (see Section 5.13)
- `debug voip` (see Section 5.14)

5.1 `debug auxiliary-files`

This command debugs loaded Auxiliary files.

Syntax

```
# debug auxiliary-files { dial-plan | user-info }
```

| Arguments | Description |
|------------------------|--|
| <code>dial-plan</code> | Debugs the dial plan (see Section 5.1.1) |
| <code>user-info</code> | Debugs the User Info file (see Section 5.1.2). |

Command Mode

Enable

5.1.1 debug auxiliary-files dial-plan

This command debugs the Dial Plan file.

Syntax

```
# debug auxiliary-files dial-plan { info | match-number < Dial Plan
Number > < Prefix Number > }
```

| Arguments | Description | |
|--------------|--|---|
| info | Displays the loaded Dial Plan file and lists the names of its defined Dial Plans. | |
| match-number | Checks whether a specific prefix number is defined in a specific Dial Plan number. If the Dial Plan is used for tags, the command also shows the tag name. | |
| | Dial Plan Number | Defines the Dial Plan in which to search for the specified prefix number. |
| | Prefix Number | Specifies the prefix number to search for in the Dial Plan. |

Note

The index number of the first Dial Plan is 0.

Command Mode

Enable

Example

Checking if the called prefix number 2000 is defined in Dial Plan 1, which is used for obtaining the destination IP address (tag):

```
# debug auxiliary-files dial-plan match-number PLAN1 2000
Match found for 4 digits
Matched prefix: 2000
Tag: 10.33.45.92
```

Displaying the loaded Dial Plan file and listing its defined Dial Plans:

```
# debug auxiliary-files dial-plan info
File Name: dialPlan.txt
Plans:
Plan #0 = PLAN1
Plan #1 = PLAN2
```

5.1.2 debug auxiliary-files user-info

This command displays the name of the User-Info file installed on the device.

Syntax

```
# debug auxiliary-files user-info info
```

| Arguments | Description |
|-----------|---|
| info | Displays the name of the User-Info file installed on the device |

Command Mode

Enable

Example

Displaying the name of the User-Info file installed on the device:

```
Mediant 1000# debug auxiliary-files user-info info
User Info File Name UIF_SBC.txt
```

5.2 debug capture

This command captures network traffic.

Syntax

```
# debug capture { trim | voip }
```

| Arguments | Description |
|-----------|-------------------|
| trim | See Section 5.2.1 |
| voip | See Section 5.2.2 |

Command Mode

Enable

5.2.1 debug capture trim

This command trims captured network traffic for USB captures.

Syntax

```
# debug capture trim { in-file [ FILE ] | offset [ HH:MM:SS ] }
```

| Arguments | Description |
|-----------|--|
| in-file | Trims captured traffic. Uses the existing file on USB storage. |
| offset | After a capture has been saved on an attached USB stick, you can trim the capture to include only a relevant time-slice. The command is useful when fetching a large capture file via SSH over a slow network connection. Offset is from the start of the capture, in hours:minutes:seconds. |

Example

Offsetting 1 hour 20 minutes from start of capture in order to trim captured USB traffic:

```
debug capture trim offset 00:01:20
```

5.2.2 debug capture voip

This command captures network traffic on VoIP network interfaces.

Syntax

```
# debug capture voip { interface | physical }
```

| Arguments | Description |
|-----------|--|
| interface | Captures network traffic on one of the VoIP sub-system network interfaces. See Section 5.2.2.1 |
| physical | Captures traffic on the wire. See Section 5.2.2.2 |

5.2.2.1 debug capture voip interface

This command captures network traffic on a VoIP network interface (VLAN).

Syntax

```
# debug capture voip interface vlan < VLAN ID > proto < protocol
filter > host < host filter > { port < port filter >
[ tftp-server < TFTP server IP address > | ftp-server < FTP server
IP address > ]}
```

➤ To start and stop the capture:

1. After typing the above command, press **Enter**.
2. To stop the capture, press **Ctrl+C**.

| Arguments | Description |
|-------------|---|
| vlan | Specifies the VLAN ID of the network interface on which to start the debug capture. |
| proto | Specifies a protocol filter: <ul style="list-style-type: none"> • all (all protocols) • arp (ARP packets) • icmp (ICMP packets) • ip (IP packets) • ipv6 (IPv6 packets) • tcp (TCP packets) • udp (UDP packets) |
| host | Specifies a host (IP address) from/to which the packets are captured. To specify all hosts, enter any. |
| port | (Optional) Specifies a port filter: 1-65535 or any (all ports). When using arp or icmp as the protocol filter, port filter cannot be used and the only valid value is any. |
| ftp-server | (Optional) Specifies the IP address of the FTP server to which the captured traffic file (in .pcap file format) is sent. If not specified, captured traffic is displayed in the CLI console. After running the command, press Ctrl+C when you want the capture to end and the captured traffic file to be sent to the server. Note: The FTP server's IP address must be accessible from one of the VoIP network interfaces for the capture file to be successfully uploaded to the server. Ping the server to make sure it's accessible. |
| tftp-server | (Optional) Specifies the IP address of the TFTP server to which the captured traffic file (in .pcap file format) is sent. If not specified, captured traffic is displayed in the CLI console. After running the command, press Ctrl+C when you want the capture to end and the captured traffic file to be sent to the server. Note: The TFTP server's IP address must be accessible |

| Arguments | Description |
|-----------|---|
| | from one of the VoIP network interfaces for the capture file to be successfully uploaded to the server. Ping the server to make sure it's accessible. |

Command Mode

Enable

Examples

Starting a debug capture on network interface VLAN 12, no host filter, and no port filter; the captured traffic is displayed in the CLI console:

```
# debug capture voip interface vlan 12 proto all host any
```

Starting a debug capture on network interface VLAN 1 with a protocol filter (IP), no host filter, and a port filter (514); the captured traffic is saved to a temporary file and is sent (when you press Ctrl+C) to the TFTP server at address 171.18.1.21:

```
# debug capture voip interface vlan 1 proto ip host any port 514
tftp-server 171.18.1.21
```


5.2.2.2 debug capture voip physical

This command captures network traffic on a physical VoIP network interface.

Syntax

```
# debug capture voip physical { cyclic-buffer | eth-lan |
get_last_capture < TFTP/FTP Server IP address > }

# debug capture voip physical target ftp user < Username >
password < Password >

# debug capture voip physical target tftp | usb
```

- To start the capture:

```
# debug capture voip physical start
```

- To stop the capture:

```
# debug capture voip physical stop { < TFTP/FTP server IP
address > | usb }
```

| Arguments | Description |
|------------------|---|
| cyclic-buffer | Continuously captures packets in a cyclical buffer. Packets are continuously captured until the <code>stop</code> command is entered. |
| eth-lan | Captures LAN frames. |
| get_last_capture | Retrieves the last captured PCAP file sent to a specified TFTP/FTP server IP address. Note: The file is saved to the device's memory (not flash) and is erased after a device reset. |
| insert-pad | Before running this command, the debug capture must be started. Inserts a PAD packet. A marked packet is shown with black background regardless of the configured coloring rules. Benefit: A marked packet can easily be located later when analyzing in a large capture file. |
| show | Displays debug status and configured rules. |
| start | Starts the capture. |
| stop | Stops the capture and sends the capture file to the specified target. The capture file is named: "debug-capture-voip-<timestamp>.pcap" |
| target | Specifies the capture storage target: <ul style="list-style-type: none"> • ftp • tftp • usb |
| user | (Only applicable if <code>ftp</code> is specified as the capture storage target) Defines the name of the FTP user. |
| password | (Only applicable if <code>ftp</code> is specified as the capture storage target) Defines the password |

| Arguments | Description |
|-----------|------------------|
| | of the FTP user. |

Command Mode

Enable

Note

- `usb` is only applicable if a USB stick is connected to the device.
- `usb` is only applicable to Mediant 5xx and Mediant 8xx devices.

Examples

- Starting a physical VoIP debug capture:


```
# debug capture voip physical eth-lan
# debug capture voip physical start
```
- Retrieving the latest capture (PCAP file) saved on a specified server.


```
# debug capture voip physical get_last_capture 10.15.7.99
```
- Specifying USB as the destination to which to send the PCAP file:


```
# debug capture voip physical target usb
```

5.3 debug debug-recording

This command enables debug recording. To collect debug recording packets, use Wireshark open-source packet capturing program. AudioCodes' proprietary plug-in files are required. They can be downloaded from www.audiocodes.com/downloads. Once downloaded, copy the plug-in files to the directory in which you installed Wireshark, as follows:

- `...\dtds\cdr.dtd:` Wireshark\dtds\
- `...\plugins\<Wireshark ver.>*.dll:` Wireshark\plugins\<Wireshark ver.>
- `...\tpncp\tpncp.dat:` Wireshark\tpncp

After starting Wireshark, type **acdr** in the 'Filter' field in order to view the debug recording messages. Note that the source IP address of the messages is always the device's OAMP IP address.

Syntax

```
# debug debug-recording < Destination IP address > { ip-trace |
port | pstn-trace | signaling | signaling-media | signaling-media-
pcm }
# debug debug-recording status
```

| Arguments | Description |
|------------------------|---|
| Destination IP address | Defines the destination IP address (IPv4) to which to send the debug recording (i.e., debug recording server). |
| ip-trace | Defines the debug recording filter type. Filters debug recording for IP network traces, using Wireshark-like expression (e.g., <code>udp && ip.addr==10.8.6.55</code>). IP traces are used to record any IP stream according to destination and/or source IP address, or port and Layer-4 protocol (UDP, TCP or any other IP type as defined by http://www.iana.com). Network traces are typically used to record HTTP. |
| port | Defines the port of the debug recording server to which to send the debug recording. |
| pstn-trace | Defines the debug recording capture type as PSTN trace. The debug recording includes ISDN and CAS traces. |
| signaling | Defines the debug recording capture type as signaling. The debug recording includes signaling information such as SIP signaling messages, Syslog messages, CDRs, and the <device>'s internal processing messages |
| signaling-media | Defines the debug recording capture type as signaling and media. The debug recording includes signaling, Syslog messages, and media (RTP/RTCP/T.38). |
| signaling-media-pcm | Defines the debug recording capture type as signaling, media and PCM. The debug recording includes SIP signalling messages, Syslog messages, media, and PCM (voice signals from and to TDM). |
| status | Displays the debug recording status. |

Note

Applicable to Mediant 500, Mediant 8xx, Mediant 1000B, Mediant 2600, Mediant 4000 and Mediant VE/SE.

Command Mode

Enable

Example

Displaying the debug recording status:

```
# debug debug-recording status
Debug Recording Configuration:
=====
Debug Recording Destination IP: 10.33.5.231
Debug Recording Destination Port: 925
Debug Recording Status: Stop
```

Logging Filter Configuration (line 0):

```
=====
Filter Type: Any
Value:
Capture Type: Signaling
Log Destination: Syslog Server
Mode: Enable
```

5.4 debug dial plan

This command checks whether a specified Dial Plan contains specific digits.

Syntax

```
# debug dial-plan < Dial Plan name > match-digits < digits to
match >
```

Command Mode

Enable

Example

Searching for digits "2000" in Dial Plan 1:

```
# debug dial-plan 1 match-digits 2000
Match succeeded for dial plan 1 and dialed number 2000. Returned
tag RmoteUser
```

5.5 debug fax

This command debugs fax modem with a debug level.

Syntax

```
# debug fax {basic | detail} [num of next sessions for debug]
```

| Arguments | Description |
|-----------|--|
| basic | Sets debug fax level to Basic. <ul style="list-style-type: none"> num of next sessions for debug (Configures the number of next sessions for debug) |
| detail | Sets debug fax level to Detail. <ul style="list-style-type: none"> num of next sessions for debug (Configures the number of next sessions for debug). |

Note

- Applicable only to FXS-supporting products.
- To disable debug fax, type:

```
# no debug fax
```

Command Mode

Enable

Example

Enabling the next 10 sessions to be traced:

```
# debug fax detail 10
```

```
FaxModem debug has been activated in DETAIL mode. The 10 next FaxModem sessions will be traced.
```

5.6 debug log

This command displays debugging messages (e.g., Syslog messages). Also displays activities performed by management users in the devices' management interfaces (CLI and Web interface).

Syntax

```
# debug log [ full ]
```

| Argument | Description |
|----------|---|
| full | (Optional) Displays more information than the regular debug messages, for example, 'SID' (Session ID) and 'S' (Syslog message sequence). Useful (for example) in determining if there's a network problem resulting from a Loss of Packets. |

Note

- When connected to the CLI through Telnet/SSH, the `debug log` command affects only the current CLI session.
- To disable logging, type `no debug log`.
 - When connected to the CLI through Telnet/SSH, the `no debug log` command affects only the current CLI session.
 - To cancel log display to *all* sessions, use the command `no debug log all`.

Command Mode

Enable

Example

Displaying debug messages:

```
Mediant 500 - MSBR# debug log
Logging started
Jun 16 13:58:54 Resource SIPMessage deleted - (#144)
Jun 16 13:58:54 (#70) SBCRoutesIterator Deallocated.
Jun 16 13:58:54 (#283) FEATURE Deallocated.
```

Displaying debug messages (full):

```
Mediant 500 - MSBR# debug log full
Logging started
Jun 16 13:59:55 local0.notice [S=707517] [SID:1192090812]
(sip_stack)(706869) Resource SIP Message deleted - (#79)
Jun 16 13:59:55 local0.notice [S=707518] [SID:1192090812]
(lgr_sbc)(706870)(#69) SBCRoutesIterator Deallocated.
Jun 16 13:59:55 local0.notice [S=707519] [SID:1192090812]
(lgr_sbc)(706871) (#282) FEATURE Deallocated.
```

5.7 debug pstn

This command activates PSTN debugging.

Syntax

```
# debug pstn
```

Note

- To deactivate PSTN debugging, type `no debug pstn`.

Command Mode

Enable

Related Commands

```
debug voip interface
```

Example

Activating PSTN debugging:

```
# debug pstn
```

5.8 debug sip

This command configures SIP debug level.

Syntax

```
# debug sip { [ < Debug Level > ] | status }
```

| Arguments | Description |
|-------------|---|
| Debug Level | Configures the SIP debug level: <ul style="list-style-type: none"> ▪ 0 = (No debug) Debug is disabled and Syslog messages are not sent. ▪ 1 = (Basic) Sends debug logs of incoming and outgoing SIP messages. ▪ 5 = (Detailed) Sends debug logs of incoming and outgoing SIP messages as well as many other logged processes. |
| status | Displays the current debug level. |

Note

- If no level is specified, level 5 is used.
- Typing `no debug sip` configures the level to **0**.

Command Mode

Enable

Example

Setting the SIP debug level to 5:

```
# debug sip 5
```


5.9 debug speedtest

This command tests the upload and download speed (in bps) to and from a specified URL, respectively.

Syntax

```
# debug speedtest set { upload | download } < URL >
# debug speedtest set upsize < Upload Transfer Bytes >
# debug speedtest run | show | stop
```

| Arguments | Description |
|--------------|---|
| set upload | Tests the upload speed to a URL (IP address or FQDN). |
| set upsize | (Optional) Defines the number of bytes (1-10000000) to upload to the specified URL for testing the upload speed |
| set download | Tests the download speed from a URL (IP address or FQDN). |
| show | Displays the test results. |
| stop | Stops the test. |
| run | Starts the test. |

Example

Testing upload speed to speedy.com:

```
# debug speedtest set upload http://www.speedy.com/speedtest
Upload URL : http://www.speedy.com/speedtest
```

```
# debug speedtest run
```

```
Starting speed test. Check results using the command "debug
speedtest show".
```

```
# debug speedtest show
```

```
Speed test results:
```

```
Upload : Complete
```

```
URL: http://www.speedy.com/speedtest
```

```
Bytes transferred: 1000000
```

```
Speed: 9.8 Mbps
```

5.10 debug syslog

This command verifies that Syslog messages sent by the device are received by the Syslog server. After you run the command, you need to check the Syslog server to verify whether it has received your Syslog message.

Syntax

```
# debug syslog < String >
```

| Arguments | Description |
|-----------|--|
| String | Defines any characters that you want to send in the Syslog message to the Syslog server. |

Command Mode

Enable

Related Commands

```
# debug syslog-server
```

Example

Verifying that a Syslog message containing "hello alan" is sent to the Syslog server:

```
# debug syslog hello alan
```

5.11 debug syslog-server

This command defines the IP address and port of the Syslog server.

Syntax

```
# debug syslog-server < IP Address > port < Port Number >
```

| Arguments | Description |
|------------|---|
| IP Address | Defines the IP address of the Syslog server. |
| port | Defines the port number of the Syslog server. |

Note

To disable Syslog server debugging, use the following command:

```
# no debug syslog-server
```

Command Mode

Enable

Example

Enabling Syslog by configuring the Syslog server:

```
# debug syslog-server 10.15.1.0 port 514
```

```
Syslog enabled to dest IP Address: 10.15.1.0 Port 514
```

5.12 debug test-call

This command initiates and terminates a call from the device to a remote destination to test whether connectivity, media, etc., are correct. Sends a SIP INVITE message and then manages the call with the call recipient.

Syntax

```
# debug test-call ip
```

- Configures a test call:

```
# debug test-call ip dial from { < Calling Number > to <
Called Number > [ dest-address < IP Address > ] [ sip-interface
< SIP Interface ID > ] | id < Test Call Table Index > }
```

- Configures a test call:

```
# debug test-call ip set called-number < Called number >
caller-id < Caller ID > calling-number < Calling number >
dest-address
< IP Address > play < Playback > sip-interfaces < SIP
Interface ID > timeout < Disconnection timeout > transport-
type
```

- Terminates a test call:

```
# debug test-call ip drop { < Calling Number > | id < Test
Call Table Index > }
```

- Displays test call configuration:

```
# debug test-call ip show
```

| Arguments | Description |
|-----------|---|
| ip | Configures and initiates a test call to an IP address. <ul style="list-style-type: none"> ▪ dial (Dials using specified parameters) <ul style="list-style-type: none"> ✓ from (Specifies the calling number): [NUMBER] (Calling number) id (uses the Test Call Rules table entry) ▪ drop (Terminates the latest outgoing test call): <ul style="list-style-type: none"> ✓ [Calling Number] (Terminates outgoing test call by number) ✓ id (Terminates outgoing test calls by table index) ▪ set (Sets test options): <ul style="list-style-type: none"> ✓ called-number (Called number) ✓ caller-id (Caller ID) ✓ calling-number (Calling number) ✓ dest-address (Target host) ✓ play (Sets playback) ✓ sip-interfaces (Sets SIP interfaces to listen on) ✓ timeout (Disconnection timeout (seconds)) ✓ transport-type (Transport type) ▪ show (Displays test call configuration) |

Command Mode

Enable

Note

- The command is applicable only to the SBC application.
- Test calls can be made with the following two recommended commands:
 - (Basic) Making a call from one phone number to another, without performing any configuration:


```
debug test-call ip dial from * to * dest-address * [sip-interface *]
```
 - (Advanced) Configuring a row in the Test Call table, and then placing a call by the row index:


```
debug test-call ip dial from id *
```

5.13 debug usb

This command debugs the USB stick connected to the device.

Syntax

```
# debug usb devices
```

| Arguments | Description |
|-----------|---|
| devices | Displays information about the USB sticks (e.g., manufacturer) connected to the device. |

Command Mode

Enable

5.14 debug voip

This command debugs voice over IP channels.

```
# debug voip { activate-channel | close-channels | dial-string |
interface | open-and-activate | open-channel | wait-for-detection}
```

| Arguments | Description |
|--------------------|--|
| activate-channel | Defines type of channel: <ul style="list-style-type: none"> ▪ analog (Activates an analog channel) ▪ digital (Activates a digital channel) ▪ virtual (Activates a virtual channel) |
| close-channels | Closes channels <ul style="list-style-type: none"> ▪ analog (FXS/FXO channel – run the <code>show system assembly</code> command to orient on the current hardware) ▪ digital (BRI/PRI channel (run the <code>show system assembly</code> command for orientation on the current hardware) ▪ virtual (Virtual channel) |
| dial-string | Sends a string of DTMF tones <ul style="list-style-type: none"> ▪ analog (FXS/FXO channel (run the <code>show system assembly</code> command for orientation on the current hardware) ▪ digital (BRI/PRI channel (run the <code>show system assembly</code> command for orientation on the current hardware) ▪ virtual (Virtual channel) |
| interface | Debugs tools related to PSTN interface <ul style="list-style-type: none"> ▪ bri (BRI debug parameters) ▪ e1-t1 (E1-T1 debug parameters) |
| open-and-activate | Opens and activates a channel <ul style="list-style-type: none"> ▪ analog (FXS/FXO channel (run the <code>show system assembly</code> command for orientation on the current hardware) ▪ digital (BRI/PRI channel (run the <code>show system assembly</code> command for orientation on the current hardware) ▪ virtual (Virtual channel) |
| open-channel | Opens channel analog/digital/virtual < CID > |
| wait-for-detection | Waits for a digit detection event |

6 Show Commands

This chapter describes the following show commands:

- `show active-alarms` (see Section 6.1)
- `show alarms-history` (see Section 6.2)
- `show clock` (see Section 6.3)
- `show high-availability` (see Section 6.4)
- `show network` (see Section 6.5)
- `show ntp` (see Section 6.6)
- `show radius` (see Section 6.7)
- `show running-config` (see Section 6.8)
- `show storage-history` (see Section 6.9)
- `show system` (see Section 6.10)
- `show tls` (see Section 6.11)
- `show users` (see Section 6.12)
- `show version` (see Section 6.13)
- `show voip` (see Section 6.14)

6.1 `show active-alarms`

This command displays active alarms.

Syntax

```
# show active-alarms
```

Command Mode

Enable

Examples

```
# show active-alarms
```

| Seq. | Source | Severity | Date | Description |
|------|-------------------------|----------|----------------------|---|
| 1. | Board#1/EthernetLink#2 | minor | 11.6.2010 , 14:19:42 | Ethernet link alarm. LAN port number 2 is down. |
| 2. | Board#1/EthernetGroup#2 | major | 11.6.2010 , 14:19:46 | Ethernet Group alarm. Ethernet Group 2 is Down. |

6.2 show alarms-history

This command displays the system alarms history.

Syntax

```
# show alarms-history
```

Command Mode

Enable

Example

```
Mediant 500# show alarms-history
```

| Seq. | Source | Severity | Date | Description |
|------|--|----------|----------------------|--------------------------|
| 1. | Interface#0/trunk#0 | critical | 24.2.2011 , 20:20:32 | Trunk LOS Alarm. |
| 2. | Board#1 state change alarm. Gateway is locked. | major | 24.2.2011 , 20:20:32 | Network element admin |
| 3. | Board#1/EthernetLink#2 port number 2 is down. | minor | 24.2.2011 , 20:20:34 | Ethernet link alarm. LAN |
| 4. | Board#1/EthernetLink#3 port number 3 is down. | minor | 24.2.2011 , 20:20:34 | Ethernet link alarm. LAN |
| 5. | Board#1/EthernetGroup#2 Ethernet Group 2 is Down. | major | 24.2.2011 , 20:20:38 | Ethernet Group alarm. |
| 6. | Board#1/EthernetGroup#3 Ethernet Group 3 is Down. | major | 24.2.2011 , 20:20:38 | Ethernet Group alarm. |

6.3 show clock

This command displays the device's time and date.

Syntax

```
# show clock
```

Command Mode

Enable

Example

```
# show clock
14:12:48 18/06/2010 (dd/mm/yyyy)
```


6.4 show high-availability

This command displays network monitor status and HA status.

Syntax

```
# show high-availability { network-monitor-status | status }
```

| Arguments | Description |
|------------------------|----------------------------------|
| network-monitor-status | Displays network monitor status. |
| status | Displays HA status. |

Command Mode

Enable

Example

```
# show high-availability status
HA Status:
Unit HA state is: StandAlone
HA Connection with other unit State is: Init
Last HA sync. action/state with other unit was: Sync. not started
```

6.5 show network

This command displays networking information.

Syntax

```
# show network { access-list | arp | dhcp clients | ether-group |
interface [ Description ] | network-dev | physical-port | route |
status }
```

| Arguments | Description |
|---------------|---|
| access-list | Displays the network access list (firewall) rules. |
| arp | Displays the ARP table entries. |
| dhcp | Displays the DHCP server leases. |
| ether-group | Displays the Ethernet Groups configured in the Ethernet Groups table. |
| interface | Displays the network IP interfaces (configured in the IP Interfaces table) with packet statistics, for example, number of transmitted packets. <ul style="list-style-type: none"> description (Displays IP network interfaces in the same format as the IP Interfaces table). |
| network-dev | Displays the Ethernet Devices configured in the Ethernet Devices table. |
| physical-port | Displays the Ethernet ports configured in the Physical Ports table. |
| route | Displays the static routes configured in the Static Routes table. |

Related Commands

Enable

Command Mode

Enable

Example

```
# show network route
Codes: C - connected, S - static
C 10.3.0.0/16 is directly connected, vlan 1, Active
C 10.99.0.0/16 is directly connected, vlan 2, Active
C 10.99.0.0/16 is directly connected, vlan 7, Active
C 10.99.0.0/16 is directly connected, vlan 8, Active
C 10.99.0.0/16 is directly connected, vlan 9, Active
C 10.99.0.0/16 is directly connected, vlan 11, Active
C 10.99.0.0/16 is directly connected, vlan 10, Active
C 10.99.0.0/16 is directly connected, vlan 12, Active
C 10.10.0.0/16 is directly connected, vlan 5, Active
```

```
C 10.10.0.0/16 is directly connected, vlan 3, Active
C 192.168.0.0/16 is directly connected, vlan 6, Active
C 192.168.0.0/16 is directly connected, vlan 4, Active
S 0.0.0.0/0 [1] via 192.168.0.111, vlan 4, Active
S 0.0.0.0/0 [1] via 10.10.0.111, vlan 3, Active
S 0.0.0.0/0 [1] via 10.3.0.1, vlan 1, Active
S 0.0.0.0/0 [1] via 10.99.0.1, vlan 2, Active
```

6.6 show ntp

This command displays NTP information.

Syntax

```
# show ntp
```

Command Mode

Enable

Example

Displaying NTP information:

```
# show ntp
Configured NTP server #1 is 10.3.1.77
NTP is synchronized, stratum 0, reference is
☺☺

** Precision          0.00000 seconds
** Root delay         0.17199 seconds
** Root dispersion    0.12752 seconds
** Reference time     dafa64cf.2c476b5f (2016-06-02 07:54:55 UTC)
** UTC offset         10800 seconds
Current local time: 2016-06-02 11:29:19
```

6.7 show radius

This command displays the status of the RADIUS servers.

Syntax

```
# show radius servers
```

Command Mode

Enable

Example

```
# show radius servers
servers 0
ip-address 10.4.4.203
auth-port 1812
auth-ha-state "ACTIVE"
acc-port 1813
acc-ha-state "ACTIVE"
servers 1
ip-address 10.4.4.202
auth-port 1812
auth-ha-state "STANDBY"
acc-port 1813
acc-ha-state "STANDBY"
```

This example shows the following fields per server:

- If the authentication port is 0, the server is not part of the redundancy server selection for authentication.
- If the accounting port is 0, the server is not part of the redundancy server selection for accounting.
- Server authentication redundancy (HA) status. ACTIVE = the server was used for the last sent authentication request.
- Server accounting redundancy (HA) status. ACTIVE = the server was used for the last sent accounting request..

6.8 show running-config

This command displays the device's configuration.

Syntax

```
# show running-config
# show running-config > < URL >
# show running-config full [ > < URL > ]
```

| Arguments | Description |
|-----------|--|
| - | Displays the device's configuration in the format of a CLI command script. You can copy and paste the displayed output in a text-based file (e.g., using Notepad), and then upload the file to another device, or the same device if you want to make configuration changes, as a CLI script file. |
| > URL | (Optional) Sends the device's configuration in CLI script format, as a file to a remote destination defined by a URL (TFTP, HTTP or HTTPS). |
| full | (Optional) Displays the device's configuration as well as the default configuration settings that were not actively set by the user. In regular mode, only configuration that is not equal to the default is displayed. |

Command Mode

Basic

Example

Sending configuration to an HTTP server:

```
# show running-config > http://10.9.9.9
```

6.9 show storage-history

This command displays the CDRs stored on the device.

Syntax

```
# show storage-history { services | unused }
```

| Argument | Description |
|-----------------|--|
| storage-history | Two options: <ul style="list-style-type: none"> ▪ services (Displays registered storage services, e.g., for CDRs) ▪ unused (Displays stored files that are not used) |

Command Mode

Enabled

Related Command

```
clear storage-history
```

6.10 show system

This command displays system information.

Syntax

```
# show system { assembly | cpu-util [ refreshing ] | cwmmp status |
fax-debug-status | feature-key | interface osn | log | uptime }
```

| Argument | Description |
|------------------|--|
| assembly | Displays information about the device's hardware assembly (slots, ports, module type, fan tray and power supply). |
| cpu-util | Displays voice CPU utilization, as a percentage <ul style="list-style-type: none"> refreshing (Refreshes the voice CPU information displayed; press CTRL+C to stop the refresh) |
| cwmmp status | Displays TR-069 status (e.g., URL of ACS). |
| fax-debug-status | Displays fax debug status. |
| feature-key | Displays the stored License Key. |
| interface osn | Displays information about the OSN module. |
| log | Displays the log history. <ul style="list-style-type: none"> -h (Displays the log history in a readable format) tail (Displays the recent history of the log messages) |
| uptime | Displays the device's uptime. |

Command Mode

Enable

Example

```
# show system assembly
```

```
Board Assembly Info:
```

```
| Slot No.          | Ports      | Module Type  |
| 1                 | 1          | E1/T1       |
| 2                 | 1-4        | FXS         |
| 3                 | 0          | Empty       |
| 4                 | 1-4        | LAN-GE      |
| 5                 | 0          | Empty       |
```

```
USB Port 1: Empty
```

```
USB Port 2: Empty
```

6.11 show tls

This command displays TLS security information (TLS Context).

Syntax

```
# show tls { certificate | contexts | trusted-root }
# show tls trusted-root { detail < index > | summary }
```

| Arguments | Description |
|--------------|--|
| certificate | Displays certificate information. |
| contexts | Displays TLS security context information. |
| trusted-root | Displays trusted certificates. <ul style="list-style-type: none"> ▪ detail (Displays a specific trusted certificate) ▪ summary (Displays all trusted certificates) |

Command Mode

Enable

Example

Displaying active TLS Contexts:

```
# show tls contexts
Context #  Name
-----  -
0         default
2         ymca

Total 2 active contexts.
Total certificate file size: 4208 bytes.
```


6.12 show users

This command displays and terminates users that are currently logged into the device's CLI. Applicable to users logged into the CLI through RS-232 (console), Telnet, or SSH.

For each logged-in user, the command displays the type of interface (console, Telnet, or SSH), user's username, remote IP address from where the user logged in, and the duration (days and time) of the session. Each user is displayed with a unique index (session ID).

Syntax

```
# show users
```

Command Mode

Basic

Note

The device can display management sessions of up to 24 hours. After this time, the duration counter is reset.

Example

Displaying all active calls:

```
# show users
[0] console      Admin      local      0d00h03m15s
[1] telnet       John       10.4.2.1   0d01h03m47s
[2]* ssh         Alex       192.168.121.234 12d00h02m34s
```

The current session from which the show command was run is displayed with an asterisk (*).

6.13 show version

This command displays the current running software and hardware version.

Syntax

```
# show version
```

Command Mode

Basic

Command Mode

Enable

6.14 show voip

This command displays VoIP-related information.

Syntax

```
# show voip { calls | channel-stats | coders-stats | cpu-stats |
dsp | e911 | ids | interface | ip-group | ldap | other-dialog |
proxy | realm | register | subscribe | tdm }
```

| Arguments | Description |
|---------------|---------------------|
| calls | See Section 6.14.1 |
| channel-stats | See Section 6.14.2 |
| coders-stats | See Section 6.14.3 |
| cpu-stats | See Section 6.14.4 |
| dsp | See Section 6.14.5 |
| e911 | See Section 6.14.6 |
| ids | See Section 6.14.7 |
| interface | See Section 6.14.8 |
| ip-group | See Section 6.14.9 |
| ldap | See Section 6.14.10 |
| other-dialog | See Section 6.14.11 |
| proxy | See Section 6.14.12 |
| realm | See Section 6.14.13 |
| register | See Section 6.14.14 |
| subscribe | See Section 6.14.15 |

6.14.1 show voip calls

This command displays active VoIP call information.

Syntax

```
# show voip calls { active | history | statistics }
```

| Arguments | Description |
|------------|----------------------|
| active | See Section 6.14.1.1 |
| history | See Section 6.14.1.2 |
| statistics | See Section 6.14.1.3 |

6.14.1.1 show voip calls active

This command displays active calls.

Syntax

```
# show voip calls active [ < Session ID > | descending | gw | sbc
| summary ]
```

| Arguments | Description |
|------------|--|
| - | Displays the total number of active calls and detailed call information. |
| Session ID | Displays detailed call information for a specific SIP session ID. |
| descending | Displays currently active calls, listed in descending order by call duration. |
| gw | Displays call information of currently active Gateway calls, listed in ascending order by call duration. |
| sbc | Displays call information of currently active SBC calls, listed in ascending order by call duration. |
| summary | Displays the total number of currently active calls (Gateway and SBC) |

Example

Displaying all active calls:

```
# show voip calls active sbc
Total Active Calls: 1000
| Session ID | Caller | Callee | Origin
| Remote IP | End Point Type | Duration | Call State
=====
| 314380675 | 1129@10.3.3.194 | 100@10.3.91.2
| Incoming | 10.3.3.194(IPG-1) | SBC | 00:05:12 | Connected
| 314380675 | 1129@10.3.3.194 | 100@10.3.91.2
| Outgoing | 10.3.3.194(IPG-2) | SBC | 00:05:12 | Connected
| 314380674 | 1128@10.3.3.194 | 100@10.3.91.2
| Incoming | 10.3.3.194(IPG-1) | SBC | 00:05:12 | Connected
...
```

6.14.1.2 show voip calls history

This command displays CDR history information.

Syntax

```
# show voip calls history { gw | sbc } [ < Session ID > ]
```

| Arguments | Description |
|------------|---|
| gw | Displays historical Gateway CDRs. |
| sbc | Displays historical SBC CDRs. |
| Session ID | (Optional) Displays historical SBC or Gateway CDRs of a specified SIP session ID. |

Example

Displaying CDR history information:

```
# show voip calls history sbc
```

6.14.1.3 show voip calls statistics

This command displays call statistics.

Syntax

```
# show voip calls statistics { gw | sbc }
# show voip calls statistics gw [ ip2tel | tel2ip ]
```

| Arguments | Description | |
|-----------|---|--|
| gw | Displays Gateway call statistics. Optional arguments: | |
| | ip2tel | Displays statistics of IP-to-Tel calls |
| | tel2ip | Displays statistics of Tel-toIP calls |
| sbc | Displays SBC call statistics (see the example below). | |

Example

Displaying SBC call statistics:

```
# show voip calls statistics sbc
SBC Call Statistics:
Active INVITE dialogs: 0
Active incoming INVITE dialogs: 0
Active outgoing INVITE dialogs: 0
Average call duration [min:sec]: 0:00
Call attempts: 0
Incoming call attempts: 0
Outgoing call attempts: 0
Established calls: 0
Incoming established calls: 0
Outgoing established calls: 0
Calls terminated due to busy line: 0
Incoming calls terminated due to busy line: 0
Outgoing calls terminated due to busy line: 0
Calls terminated due to no answer: 0
Incoming calls terminated due to no answer: 0
Outgoing calls terminated due to no answer: 0
Calls terminated due to forward: 0
Incoming calls terminated due to forward: 0
Outgoing calls terminated due to forward: 0
Calls terminated due to resource allocation failure: 0
Incoming calls terminated due to resource allocation failure: 0
Outgoing calls terminated due to resource allocation failure: 0
Calls terminated due to media negotiation failure: 0
Incoming calls terminated due to media negotiation failure: 0
Outgoing calls terminated due to media negotiation failure: 0
Calls terminated due to general failure: 0
Incoming calls terminated due to general failure: 0
Outgoing calls terminated due to general failure: 0
Calls abnormally terminated: 0
Incoming calls abnormally terminated: 0
```

Outgoing calls abnormally terminated: 0

6.14.2 show voip channel-stats

This command displays statistics associated with a specific VoIP channel.

Syntax

```
# show voip channel-stats { analog | channel-count | digital |
jitter-threshold | pl | pl-threshold | rtt-threshold | virtual }
```

| Arguments | Description |
|------------------|--|
| analog | Displays an analog channel's statistics (FXS or FXO). <ul style="list-style-type: none"> channel number (0-255; run the command <code>show system assembly</code> to facilitate defining this argument) number of channels (1-256) |
| channel-count | Displays the number of active voice channels. |
| digital | Displays a digital channel's statistics (E1/T1 or BRI). <ul style="list-style-type: none"> channel number (0-255; run the command <code>show system assembly</code> to facilitate defining this argument) number of channels (1-256) |
| jitter-threshold | Displays the number of analog channels, digital channels, and virtual channels on which jitter occurred that exceeded the threshold you defined (in the range 0-65535). |
| pl | Displays the number of analog channels, digital channels, and virtual channels on which PL (packet loss) occurred. |
| pl-threshold | Displays the number of analog channels, digital channels, and virtual channels on which PL (packet loss) occurred that exceeded the threshold you defined (in the range 0-65535). |
| rtt-threshold | Displays the number of analog channels, digital channels, and virtual channels on which the RTT (Round Trip Time) exceeded the threshold you defined (in the range 0-65535). |
| virtual | Displays a virtual channel's statistics. <ul style="list-style-type: none"> channel number (0-255; run the command <code>show system assembly</code> to facilitate defining this argument) number of channels (1-256) |

6.14.3 show voip coders-stats

This command displays the number and percentage of active channels using each audio coder.

Syntax

```
# show voip coders-stats
```

Example

Showing that 67 channels (25.18%) of the 266 active channels are using the G.729e coder, 76 (28.57%) are using the G.726 coder, and 123 (46.24%) are using the G.722 coder:

```
# show voip coders-stats
There are 266 active channels.
Coder      Number of Channels      Percentage
-----
G729e      67                       25.18
G726       76                       28.57
G722       123                      46.24
```

6.14.4 show voip cpu-stats

This command displays the device's CPU percentage use.

Syntax

```
# show voip cpu-stats
```

Example

Displaying CPU percentage use:

```
# show voip cpu-stats
CPU percentage: 47%
```

6.14.5 show voip dsp

This command displays DSP information.

6.14.5.1 show voip dsp perf

This command displays performance monitoring of DSP data.

Syntax

```
# show voip dsp perf
```

Example

Displaying performance monitoring of DSP data:

```
# show voip dsp perf
```

```
DSP Statistics (statistics for 144 seconds):
```

```
Active DSP resources: 0
```

```
Total DSP resources: 76
```

```
DSP usage : 0
```

6.14.5.2 show voip dsp status

This command displays the current DSP status.

Syntax

```
# show voip dsp status
```

Example

Displaying the current DSP status:

```
# show voip dsp status
```

```
Group:0 DSP firmware:624AE3 Version:0660.07 - Used=0 Free=72  
Total=72
```

```
  DSP device  0:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  1:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  2:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  3:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  4:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  5:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  6:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  7:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  8:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device  9:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device 10:  Active    Used= 0   Free= 6   Total= 6
```

```
  DSP device 11:  Active    Used= 0   Free= 6   Total= 6
```

```
Group:1 DSP firmware:204IM Version:0660.07 - Used=0 Free=8 Total=8
```

```
  DSP device 12:  Active    Used= 0   Free= 4   Total= 4
```

```
  DSP device 13:  Active    Used= 0   Free= 4   Total= 4
```



```
Group:2 DSP firmware:204IM Version:0660.07 - Used=0 Free=4 Total=4
  DSP device 14: Active Used= 0 Free= 4 Total= 4
Group:4 DSP firmware:204IM Version:0660.07 - Used=4 Free=0 Total=4
  DSP device 15: Active Used= 4 Free= 0 Total= 4
```

6.14.6 show voip e911

This command displays the ELIN number per E911 caller and the time of call.

Syntax

```
# show voip e911
```

Note

Applicable to Mediant 500, Mediant 8xx, Mediant 1000B, Mediant 2600, and Mediant 4000.

6.14.7 show voip ids

This command displays the Intrusion Detection System (IDS) blacklist of remote hosts (IP addresses / ports) considered malicious.

Syntax

```
# show voip ids { blacklist active | active-alarm }
# show voip ids active-alarm { all | match < ID > rule < ID > }
```

| Arguments | Description |
|------------------|---|
| active-alarm | Displays all active blacklist alarms: <ul style="list-style-type: none"> ▪ all (Displays all active alarms) ▪ match (Displays active alarms of an IDS matched ID and rule ID) |
| blacklist active | Displays blacklisted hosts. |

Related Commands

```
ids policy
ids rule
```

Example

- Displaying the IDS blacklist:

```
# show voip ids blacklist active
Active blacklist entries:
10.33.5.110(NI:0) remaining 00h:00m:10s in blacklist
```

Where *SI* is the SIP Interface, and *NI* is the Network interface.

- Displaying the blacklist of all active IDS alarms:

```
# show voip ids active-alarm all
IDSMatch#0/IDSRule#1: minor alarm active.
```

- Displaying details regarding an active IDS alarm of the specified match and rule IDs:

```
# show voip ids active-alarm match 0 rule 1
IDSMatch#0/IDSRule#1: minor alarm active.
- Scope values crossed while this alarm is active:
  10.33.5.110(SI0)
```

6.14.8 show voip interface

This command displays information (basic configuration, status and Performance Monitoring) of a specified telephony interface (E1/T1, BRI or FXS/FXO).

Syntax

```
# show voip interface { e1-t1 | bri | fxs-fxo } < Module > / < Port >
```

| Arguments | Description |
|-----------|---|
| e1-t1 | Displays information on a specified E1/T1 interface. |
| bri | Displays information on a specified BRI interface. |
| fxs-fxo | Displays the current status, main PM parameters and main configuration parameters to a specific analog interface (FXS or FXO) |
| Module | Specifies the module slot index as shown on the front panel |
| Port | Specifies the module's analog port number (FXS/FXO) or trunk port number (E1/T1 or BRI) to display. |

Command Mode

Enable

Related Commands

```
(config-voip)# interface e1-t1
(config-voip)# interface bri
```

Note

- Parameters displayed depend on the PSTN protocol type.
- Applicable to devices supporting analog and/or digital PSTN interfaces.

Example

Displaying information of the E1/T1 interface of trunk port 1 of trunk module 3:

```
# show voip interface e1-t1 3/1
show voip interface e1-t1 3/1
-----
module/port:      3/1
trunk number:    0
protocol:        t1_transparent
state:           not active
alarm status:    LOS 1, LOF 0, RAI 0, AIS 0, RAI_CRC 0
loopback status: no loop
send alarm status: no alarm
```

```
main performance monitoring counters collected in the last 470
seconds:
```

```
BitError:          0          EBitErrorDetected:    0
CRCErrorReceived:  0          LineCodeViolation:    0
ControlledSlip:   0          ControlledSlipSeconds: 0
ErroredSeconds:   0          BurstyErroredSeconds: 0
UnavailableSeconds: 470      PathCodingViolation:  0
LineErroredSeconds: 0        SeverelyErroredSeconds: 0
SeverelyErroredFramingSeconds: 0
```

```
basic configuration:
```

```
framing:          T1_FRAMING_ESF_CRC6
line-code:        B8ZS
clock-master:     CLOCK_MASTER_OFF
clock-priority:   0
trace-level:      no-trace
```

6.14.9 show voip ip-group

This command displays the following QoS metrics per IP Group:

- QoE profile metrics per IP Group and its associated Media Realm on currently established calls such as MOS, jitter, packet loss, and delay. Metrics are displayed as average amounts.
- Bandwidth Profile (BW) metrics for Tx and Rx traffic per IP Group and/or Media Realm. Metrics are displayed with a status color for each specific port.
- QoE profile metrics for the remote (far-end) such as MOS, jitter, packet loss, and delay. Each metric is displayed with a specific color.
- Group MSA metrics for the IP Group and the Media Realm. Metrics are displayed as an aggregated value.

Syntax

```
# show voip ip-group <IP Group table row index> media-statistics
```

| Argument | Description |
|--------------------------|------------------------------------|
| IP Group table row index | Specifies the IP Group to display. |

Example

Displaying QoS metrics of IP Group configured in row index 0:

```
# show voip ip-group 0 media-statistics
IPGroup 0. BWProfile: -1, QoEProfile: -1
-----
MSA: 0
Averages: MOS 0 Remote MOS 0 Delay 0 Remote Delay 0 Jitter 0
Remote Jitter 0
Fraction loss tx 0 Fraction loss rx 0
Packet sent 0 Packet received 0
Audio Tx BW 0, Audio Tx Status Green
Audio Rx BW 0, Audio Rx Status Green
Total Tx BW 0, Total Tx Status Green
Total Rx BW 0, Total Rx Status Green
Video Tx BW 0, Video Tx Status Green
Video Rx BW 0, Video Rx Status Green
MSA color Gray MSA remote color Gray
MOS color Gray remote MOS color Gray
Delay color Gray remote Delay color Gray
PL color Gray remote PL color Gray
Jitter color Gray remote Jitter color Gray
color is not relevant
Media Realm -1. BWProfile -1, QoEProfile: -1
```

6.14.10 show voip ldap

This command displays the number of 'internal AD search requests', i.e., routings requiring information from the AD, including requests answered via the cache and directly from the AD. Routing requests are stored every 15 minutes. The last 96 intervals (24h) are stored.

Syntax

```
# show voip ldap cache-hits-pm | print-cache { group < Group
Matrix Index > } | print-cache-entry { group < group index > } |
print-cache-nums | searches-pm | timeout-pm
```

| Arguments | Description |
|-------------------|--|
| cache-hits-pm | Displays the number of responses answered by the cache in each interval. |
| print-cache | Displays the cache (by group). |
| print-cache-entry | Displays a cache entry (by key and group). |
| print-cache-nums | Displays the number of entries and aged entries in the cache. |
| searches-pm | Displays performance monitoring results for searches. |
| timeout-pm | Displays performance monitoring results for searches. |

Command Mode

Enable

Example

Displaying the the number of responses answered by the cache in each interval:

```
# show voip ldap cache-hits-pm
server 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 server 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Displaying the cache (by group):

```
# show voip ldap print-cache
print cache
servers' group number 0 Hash size 0 aged 0
servers' total Hash size 16384
servers' group number 1 Hash size 0 aged 0
```

Displaying the cache (by key and group):

```
# show voip ldap print-cache-entry
servers' group number 0 Hash size 0 aged 0
servers' total Hash size 16384
servers' group number 1 Hash size 0 aged 0
```

6.14.11 show voip other-dialog

This command displays the number of current incoming and outgoing SIP dialogs (e.g., REGISTER), except for INVITE and SUBSCRIBE messages.

Syntax

```
# show voip other-dialog statistics
```

Note

Applicable only to the SBC application.

Example

```
# show voip other-dialog statistics
SBC other Dialog Statistics:
Active other dialogs: 0
Active incoming other dialogs: 0
Active outgoing other dialogs: 0
```

6.14.12 show voip proxy

This command displays the status of Proxy Sets. The status ("OK" or "FAIL") indicates IP connectivity with the proxy server.

Syntax

```
# show voip proxy sets status
```

Example

Displaying status of Proxy Sets:

```
# show voip proxy sets status
```

```
Active Proxy Sets Status
ID      IP ADDRESS          STATUS
0       Not Used(--)       --
1       10.8.6.31(10.8.6.31) OK
2       10.8.2.19(10.8.2.19) OK
```

6.14.13 show voip realm

This command displays Media Realms and Remote Media Subnets.

Syntax

- Displaying Media Realms:

```
# show voip realm < Media Realm row index > statistics
```

- Displaying Remote Media Subnets:

```
# show voip realm < Media Realm row index > remote-media-subnet < Remote Media Subnet row index > statistics
```

6.14.14 show voip register

This command displays registration status of users.

Syntax

```
# show voip register { account | board | db sbc | ports |
suppserv gw | user-info }
# show voip register account { gw | sbc }
```

| Argument | Description |
|-------------|---|
| account | Displays registration status of user Accounts (Accounts table). <ul style="list-style-type: none"> ▪ gw (Gateway accounts) ▪ sbc (SBC accounts) |
| board | Displays registration status for the entire gateway. |
| db sbc | Displays SBC users registered with the device (SBC User Information table). <ul style="list-style-type: none"> • list (Displays the status of all registered SBC users showing their AOR and Contact) • user < AOR > (Displays detailed information about a specific registered SBC user, including the IP Group to which the user belongs): <p>Active:YES = user was successfully registered. Active:NO = user was registered and is waiting for approval.</p> <p>Note: Only applicable to the SBC application.</p> |
| ports | Displays registration status of the devices' ports. <p>Note: Only applicable to the Gateway application.</p> |
| suppserv gw | Displays the number of users in the Supplementary Services table. <ul style="list-style-type: none"> • list (Displays detailed information about users, including registration status (REGISTERED / NOT REGISTERED)). <p>Note: Only applicable to the Gateway application.</p> |
| user-info | Displays registration status of users in the User Info table. <ul style="list-style-type: none"> ▪ gw (Displays total number of Gateway users) <ul style="list-style-type: none"> ✓ list (Displays detailed information about users, including registration status - REGISTERED / NOT REGISTERED). ▪ sbc (Displays total number of SBC users) |

| Argument | Description |
|----------|--|
| | ✓ list (Displays detailed information about users, including registration status - REGISTERED / NOT REGISTERED). |

Example

- Displaying registration status of SBC users of AOR "2017":

```
# show voip register db sbc user 2017
*** SBC Registered Contacts for AOR '2017' ***
sip:2017@10.8.2.225:5080;expire=90; Active: YES; IPG#4;
ResourceID#(#983)
```

- Displaying port registration status:

```
# show voip register ports

*** Ports Registration Status ***

Gateway      Port      Status
=====
Module 3     Port 1    FXO      REGISTERED
-----
Module 3     Port 2    FXO      REGISTERED
-----
Module 3     Port 3    FXO      REGISTERED
-----
Module 3     Port 4    FXO      NOT REGISTERED
-----
Module 5     Port 1    FXS      NOT REGISTERED
-----
Module 5     Port 2    FXS      NOT REGISTERED
-----
Module 5     Port 3    FXS      NOT REGISTERED
-----
Module 5     Port 4    FXS      REGISTERED
```

- Displaying detailed information about users in the Supplementary Services table:

```
# show voip register suppserv gw list
*** GW Supp Serv Users Registration Status ***
Index  Type      Status      Contact
=====
1      EndPoint  NOT REGISTERED  sip:4000@10.15.7.96:5060
```

6.14.15 show voip subscribe

This command displays active SIP SUBSCRIBE dialog sessions.

Syntax

```
# show voip subscribe { list | statistics }
# show voip subscribe list [ < Session ID > | descending | summary ]
```

| Argument | Description |
|------------|---|
| list | Displays SUBSCRIBE dialog information. One of three options can be selected: <ul style="list-style-type: none"> ▪ < Session ID > (Displays detailed information for the specified Session ID). ▪ descending (Displays SUBSCRIBE dialogs sorted in descending order by call duration). ▪ summary (Displays a summary of SUBSCRIBE dialogs). |
| statistics | Displays SUBSCRIBE dialog statistics including incoming and outgoing SUBSCRIBEs. |

Example

Displaying a summary of active SUBSCRIBE dialogs:

```
# show voip subscribe statistics
SBC SUBSCRIBE Dialog Statistics:
Active SUBSCRIBE dialogs: 4
Active incoming SUBSCRIBE dialogs: 6
Active outgoing SUBSCRIBE dialogs: 8
```

7 Clear Commands

This chapter describes the following clear commands:

- clear alarms-history (see Section 7.1)
- clear debug-file (see Section 7.2)
- clear qos (see Section 7.3)
- clear storage-history (see Section 7.4)
- clear system-log (see Section 7.5)
- clear user (see Section 7.6)
- clear voip (see Section 7.7)

7.1 clear alarms-history

This command deletes the Alarms History table.

Syntax

```
# clear alarms-history
```

Command Mode

Enable

7.2 clear debug-file

This command deletes the debug file (core dump).

Syntax

```
# clear debug-file
```

Command Mode

Enable

7.3 clear qos

This command deletes counter data related to quality of service.

Syntax

```
# clear qos counters
```

Command Mode

Enable

7.4 clear storage-history

This command deletes the locally stored CDRs.

Syntax

```
# clear storage-history < Service Name >
```

| Argument | Description |
|--------------|--|
| Service Name | The name of the service. To view services, run the <code>show storage-history services</code> command. Currently supported service: <code>cdr-storage-history</code> Includes the following arguments: |
| all | Deletes all stored CDR files |
| unused | Deletes unused stored CDR files |

Command Mode

Enable

Related Commands

```
show storage-history services
```

Example

- Deleting all stored CDR files:

```
# clear storage-history cdr-storage-history all
```
- Deleting all unused stored CDR files:

```
# clear storage-history cdr-storage-history unused
```

7.5 clear system-log

This command deletes the system log.

Syntax

```
# clear system-log
```

Command Mode

Enable

7.6 clear user

This command terminates CLI users who are currently logged in through RS-232 (console), Telnet, or SSH. When run, the command drops the Telnet/SSH session or logs out the RS-232 session, and displays the login prompt.

Syntax

```
# clear user < Session ID >
```

| Argument | Description |
|------------|---|
| Session ID | Unique identification of each currently logged in CLI user. Allows you to end the active CLI session of a specific CLI user. You can view session IDs by running the show users command. |

Note

The CLI session from which the command is run cannot be terminated.

Command Mode

Enable

Related Commands

```
# show users
```

Example

Ending the CLI session of a specific user:

```
# clear user 1
```

7.7 clear voip

This command deletes VoIP-related information.

Syntax

```
# clear voip { calls | register | statistics }
```

| Arguments | Description |
|------------|-----------------------------------|
| calls | See Section 7.7.1 |
| register | See Section 7.7.2 |
| statistics | See Section 7.7.3 |

Command Mode

Enable

7.7.1 clear voip calls

This command deletes all active calls.

Syntax

```
# clear voip calls [ < Session ID > ]
```

| Arguments | Description |
|------------|--|
| - | If Session ID isn't specified, <i>all</i> active VoIP calls are cleared. |
| Session ID | (Optional) If Session ID <i>is</i> specified, the specified call is cleared. |

Command Mode

Enable

Related Commands

```
show voip calls active
```

Example

Displaying and then clearing VoIP calls:

```
# show voip calls
Total Active Calls: 1
| Session ID      | Caller          | Callee          | Origin
| Remote IP      | End Point Type | Duration        | Call State
=====
| 326433737      | 3005           | 2000           | Outgoing
| 10.8.6.36      | FXS-3/3       | 00:00:06      | Connected

# clear voip calls 326433737
1 Active Calls were Manually disconnected
```

7.7.2 clear voip register

This command deletes SBC users registered from the device's registration database.

Syntax

```
# clear voip register db sbc user < AOR >
# clear voip register db sbc ip-group < ID or Name >
```

| Arguments | Description |
|------------|---|
| AOR | Specifies the Address of Record (AOR) of the user (user part or user@host). |
| ID or name | Specifies an IP Group (i.e., deletes all registered users belonging to the IP Group). |

Command Mode

Enable

Note

The command is applicable only to the SBC application.

Example

Clearing John@10.33.2.22 from the registration database:

```
# clear voip register db sbc user John@10.33.2.22
```

7.7.3 clear voip statistics

This command deletes calls statistics.

Syntax

```
# clear voip statistics
```

Command Mode

Enable

Example

Clearing calls statistics:

```
# clear voip statistics
```


8 Maintenance Commands

This chapter describes the following maintenance-related commands:

- admin register (see Section 8.1)
- copy (see Section 8.2)
- dir (see Section 8.3)
- erase (see Section 8.4)
- nslookup (see Section 8.5)
- ping (see Section 8.6)
- pstn nfas-group-switch-activity (see Section 8.7)
- reload (see Section 8.8)
- srd-view (see Section 8.9)
- telnet (see Section 8.10)
- traceroute (see Section 8.11)
- usb (see Section 8.12)
- write (see Section 8.13)
- write factory (see Section 8.14)
- write-and-backup (see Section 8.15)

8.1 admin register

This command registers (or unregisters) users with a proxy server.

Syntax

```
# admin register { accounts | gw | ports | suppserv | userinfo }
```

- Registration by Accounts:

```
# admin register accounts < Account Index >
```

- Registering Gateway:

```
# admin register gw
```

- Registration per physical port:

```
# admin register ports < Module Number > < Port Number >
```

- Registration by extension number:

```
# admin register suppserv < Extension Number >
```

- Registration by User Info table:

```
# admin register userinfo { gw | sbc } < Local User >
```

| Argument | Description |
|----------|---|
| accounts | Registers user Accounts, configured in the Accounts table. |
| gw | Registers the device as a single entity (Gateway). |
| ports | Registers the device's ports. You need to specify the module number and port number. |
| suppserv | Registers an FXS endpoint by phone number and BRI line extensions configured in the Supplementary Services table. |
| userinfo | Registers users configured in the User Info table. |

Note

To unregister, use the below command:

```
# admin unregister { accounts | gw | ports | suppserv | userinfo }
```

Command Mode

Enable

Example

Registering port 1 located on module 3:

```
# admin register ports 3 1
```

```
Registering module 3 port 1 (200)
```

8.2 copy

This command downloads files to the device, or uploads files from the device.

Syntax

```
# copy { aux-package | call-progress-tones | cas-table | cli-script | debug-file | dial-plan | firmware | prerecorded-tones | storage-history | tls-cert | tls-private-key | tls-root-cert | user-info | voice-configuration | voice-prompts | web-favicon } { from | to } { < URL > | console | usb:///< File Name > }
```

| Argument | Description |
|---------------------|--|
| File Type | |
| aux-package | <p>Defines the type of file as an auxiliary package file, allowing you to download or upload a batch of auxiliary files, using a TAR (Tape ARchive) file (.tar).</p> <p>The TAR file can contain any number and type of Auxiliary files, for example, a Dial Plan file and a CPT file.</p> <p>Note: The file can be uploaded or downloaded (see arguments 'from' and 'to' below).</p> |
| call-progress-tones | <p>Defines the type of file as a Call Progress Tones (CPT) file.</p> <p>Note: The file can only be uploaded to the device (see the argument 'from' below).</p> |
| cas-table | <p>Defines the type of file as a Channel Associated Signaling (CAS) table file.</p> <p>Note: The file can only be uploaded to the device (see the argument 'from' below).</p> |
| cli-script | <p>Defines the type of file as a CLI script file.</p> <p>Note:</p> <ul style="list-style-type: none"> The file can be uploaded or downloaded (see arguments 'from' and 'to' below). Copying to or from USB is applicable only to Mediant 5xx and Mediant 8xx. |
| debug-file | <p>Defines the type of file as a debug file and copies the file from the <device> to a destination. The debug file contains the following information:</p> <ul style="list-style-type: none"> Exception information, indicating the specific point in the code where the crash occurred and a list of up to 50 of the most recent SNMP alarms that were raised by the <device> before it crashed. Latest log messages that were recorded prior to the crash. Core dump. The core dump is included only if core dump generation is enabled, no IP address has been configured, and the <device> has sufficient memory on its flash memory. <p>May include additional application-proprietary debug</p> |

| Argument | Description |
|------------------------|--|
| | information. The debug file is saved as a zipped file with the following file name: "debug_<device name>_ver_<firmware version>_mac_<MAC address>_<date>_<time>". For example, debug_acMediant_ver_700-8-4_mac_00908F099096_1-03-2015_3-29-29. |
| dial-plan | Defines the type of file as a Dial Plan file. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| firmware | Defines the type of file as a firmware file (.cmp). Note: After the .cmp file is loaded to the device, it's automatically saved to the device's flash memory with a device reset. |
| prerecorded-tones | Defines the type of file as a Prerecorded Tones (PRT) file. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| storage-history | Defines the type of file as a locally stored Call Detail Record (CDR) file. Define the name of the service. To view services, run the command <code>show storage-history services</code> . Currently supported service: <code>cdr-storage-history</code> |
| tls-cert | Defines the type of file as a TLS certificate file. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| tls-private-key | Defines the type of file as a TLS private key file. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| tls-root-cert | Defines the type of file as a TLS trusted root certificate file. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| user-info | Defines the type of file as a User Info file. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| voice-configuration | Defines the type of file as a configuration file (.ini). Note: The file can be uploaded or downloaded (see arguments 'from' and 'to' below). |
| voice-prompts | Defines the type of file as a Voice Prompts (VP) file. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| web-favicon | Defines the type of file as an icon file associated with the device's URL saved as a favorite bookmark on your browser's toolbar. Note: The file can only be uploaded to the device (see the argument 'from' below). |
| Download/Upload | |
| from | Downloads a file to the device. |
| to | Uploads a file from the device to a specified destination. |

| File Location | |
|---------------------|---|
| URL | <p>Defines the URL from which / to which to upload / download the file. Can be:</p> <ul style="list-style-type: none"> ▪ HTTP ▪ HTTPS ▪ TFTP |
| console | <p>Displays the current .ini configuration file on the CLI console.</p> <p>Note: Applicable only to the .ini configuration file (voice-configuration).</p> |
| usb:///< file name> | <p>Uploads the file from a USB stick, connected to the device, to the device, or downloads the file from the device to a USB stick connected to the device.</p> <p>Note: Applicable only to devices that provide a USB port interface.</p> |

Command Mode

Enable

Related Commands

```
erase
dir
write
```

Note

- When you load a file to the device, you must run the `write` command to save the file to flash memory, else the file is deleted when the device is reset or powered off.
- For more information on the different file types, refer to the *User's Manual*.

Example

- Copying firmware file from an HTTP server:


```
# copy firmware from
http://192.169.11.11:80/SIP_F7.20A.260.002.cmp
Copying file...
done.
Restarting...
```
- Copying the voice configuration file to the CLI console:


```
# copy voice-configuration to console
```
- Auxilliary file batch:


```
# copy myauxfiles.tar from http://www.exmample.com/auxiliary
```
- Copying CLI-based configuration from TFTP server:


```
# copy cli-script from tftp://192.168.0.3/script1.txt
```
- Upgrading the device's firmware from a source URL file:


```
# copy firmware from http://www.exmample.com/firmware.cmp
```
- Copying the dial plan file:

```
copy dial-plan from http://10.4.2.2/MyHistoryFiles/
```

After the `copy` command is run, progress information is displayed. Example:

| % Total | % Received | % Xferd | Average Dload | Speed Upload | Time Total | Time Spent | Time Left | Current Speed |
|-----------|------------|---------|---------------|--------------|------------|------------|-----------|---------------|
| 100 29.2M | 100 29.2M | 0 0 | 939k | 0 | 0:00:31 | 0:00:31 | --:-- | 945k |

where

- % is the percentage of total bytes downloaded and uploaded; downloaded is displayed only when downloading a file (i.e., `copy from` command)
- **Total** is the total bytes downloaded and uploaded.
- % is the percentage of downloaded bytes, for the `copy from` command.
- **Received** is the currently downloaded bytes, for the `copy from` command.
- % is the percentage of uploaded bytes, for the `copy to` command.
- **Xferd** is the currently uploaded bytes, for the `copy to` command.
- **Average Dload** is the average download speed, in bytes/sec, for the `copy from` command.
- **Speed Upload** is the average upload speed, in bytes/sec, for the `copy to` command.
- **Time Spent** is the elapsed time.
- **Time Left** is the time remaining for the file upload/download to complete.
- **Current Speed** is the current upload/download speed, in bytes/sec.

8.3 dir

This command displays the device's current auxiliary files directory.

Syntax

```
# dir
```

Command Mode

Enable

Example

Displaying the device's current auxiliary files directory:

```
# dir
directory listing:
call-progress-tones [usa_tones_13.dat] 9260 Bytes
cas-table [Earth_Calling.dat] 43852 Bytes
tls-private-key [pkey.pem] 940 Bytes
tls-cert [server.pem] 643 Bytes
```

8.4 erase

This command deletes an Auxiliary file.

Syntax

```
# erase < Auxiliary File >
```

| Arguments | Description |
|----------------|--|
| Auxiliary File | Specifies the <i>type</i> of Auxilliary file to be erased. |

Note

- View files using the `dir` command.
- To make sure the file type is correctly entered, copy it from the `dir` command output.
- The `erase` command only deletes the file from the device's RAM (and from the device's current usage). To delete the file permanently (from flash memory), enter the `write` command after issuing the `dir` command.

Command Mode

Enable

Related Commands

```
dir  
write
```

Example

- Viewing Auxilliary files:

```
# dir  
directory listing:  
call-progress-tones [usa_tones_13.dat] 9260 Bytes  
cas-table [Earth_Calling.dat] 43852 Bytes  
tls-private-key [pkey.pem] 940 Bytes  
tls-cert [server.pem] 643 Bytes
```

- Erasing the CPT file from flash memory:

```
# erase call-progress-tones  
# write
```

8.5 nslookup

This command deletes queries the Domain Name System (DNS) to obtain domain name mapping or IP address mapping.

Syntax

```
nslookup < Hostname > [ source voip interface vlan < VLAN ID> ] [
type { a | aaaa | naptr | srv }
```

| Argument | Description |
|-------------------------------|--|
| Hostname | Defines the host name. |
| source voip interface vlan | (Optional) Specifies a VLAN ID (1 -3999). |
| type | (Optional) Specifies the type of DNS: <ul style="list-style-type: none"> ▪ a (Use a Host address) ▪ aaaa (Use an IPv6 Address) ▪ naptr (Use NAPTR - Naming Authority PoinTeR) ▪ srv (Use Server selection) |

Note

The DNS server must be configured for this command to function. The DNS server can be configured using:

- Internal DNS table: configure network > dns dns-to-ip
- Internal SRV table : configure network > dns srv2ip
- IP Interfaces table: configure network > interface network-if

Command Mode

Enable

Example

Looking up the IP address of Google:

```
# nslookup google.com
google.com resolved to 216.58.213.174
```


8.6 ping

This command deletes pings a remote destination to check connectivity.

Syntax

```
# ping { < IPv4 Address > | ipv6 < IPv6 Address > } [ source voip
interface { vlan < VLAN ID > | name < Interface Name > } ] [ repeat
< Echo Requests > ] [ size < Payload Size > ]
```

| Argument | Description |
|--------------------------|---|
| IPv4 Address | Configures the IPv4 IP address in dotted-decimal notation. |
| IPv6 Address | Configures an IPv6 address as X:X::X:X. |
| source voip interface | (Optional) Configures the interface from where you want to ping. This can be one of the following: <ul style="list-style-type: none"> vlan (defines the VLAN ID) name (defines the IP network interface name) |
| repeat | (Optional) Configures the number (1-300) of echo requests. |
| size | (Optional) Configures the payload size (0-max packet size). |

Command Mode

Enable

Example

- Sending 3 ICMP packets with 555 bytes payload size to 10.4.0.1 via interface VLAN 1:

```
# ping 10.4.0.1 source voip interface vlan 1 repeat 3 size 555
PING 10.4.0.1 (10.4.0.1): 555 data bytes
563 bytes from 10.4.0.1: icmp_seq=0 ttl=255 time=1.3 ms
563 bytes from 10.4.0.1: icmp_seq=1 ttl=255 time=1.1 ms
563 bytes from 10.4.0.1: icmp_seq=2 ttl=255 time=1.2 ms
--- 10.4.0.1 ping statistics ---
3 packets transmitted, 3 packets received, 0 packet loss
round-trip min/avg/max = 1.1/1.2/1.3 ms
```

- Pinging an IPv6 destination address).

```
ping ipv6 2001:15::300
```

8.7 pstn nfas-group-switch-activity

This command deletes initiates a manual switchover between D-channels (primary and backup) pertaining to the same Non-Facility Associated Signaling (NFAS) group.

Syntax

```
# pstn nfas-group-switch-activity [ NFAS Group Number ]
```

| Arguments | Description |
|-------------------|---|
| NFAS Group Number | Specifies the NFAS group number (1-12). |

Note

Applicable only to Mediant 500, Mediant 8xx and Mediant 1000B.

Command Mode

Enable

Example

```
# pstn nfas-group-switch-activity 2
```

8.8 reload

This command deletes resets the device with or without saving the configuration to flash memory.

Syntax

```
# reload { if-needed | now | without-saving [ in < Minutes > ] }
```

| Arguments | Description |
|----------------|---|
| if-needed | Resets the device only if you have configured a parameter that requires a device reset to take effect. |
| now | Resets the device and saves the configuration (including Auxiliary files) to flash memory. |
| without-saving | Resets the device without saving the configuration to flash memory. <ul style="list-style-type: none"> in (Specifies the number of minutes that will lapse before the device is reset. Use before making changes to sensitive settings. If your changes cause the device to lose connectivity, wait for the device to restart with the previous working configuration. To cancel the timed restart, use the <code>no reload</code> command). |

Default

When the command is run, the configuration is saved to flash memory.

Command Mode

Enable

Related Command

write

8.9 srd-view

This command access a specific SRD (tenant) view. To facilitate configuration of the Multi-Tenancy feature through the CLI, the administrator can access a specific tenant view. Once in a specific tenant view, all configuration commands apply only to that specific tenant and the tenant's name (SRD name) forms part of the CLI prompt. Only table rows (indexes) belonging to the viewed tenant can be modified. New table rows are automatically associated with the viewed tenant (i.e., SRD name).

Syntax

```
# srd-view < SRD Name >
```

| Argument | Description |
|----------|--------------------------------------|
| SRD Name | Defines the tenant view by SRD name. |

Command Mode

Enable

Note

To exit the tenant view, enter the following command:

```
# no srd-view
```

Related Commands

```
voip-network srd
```

Example

Accessing the 'itsp' tenant view:

```
# srd-view itsp
(srd-itsp)#
```

8.10 telnet

This command invokes a Telnet session from the device towards a remote host for remote management. A remote administrator can access the device's CLI from the WAN leg while performing the full authentication process. The administrator can then invoke Telnet sessions towards other devices in the LAN to manage them. No special pin-holes or forwarding rules need be declared to manage them.

Syntax

```
# telnet < Address > < Port > interface vlan < VLAN ID >
```

| Arguments | Description |
|----------------|--|
| Address | Remote host IP address. |
| Port | (Optional) Remote host port number. |
| interface vlan | (Optional) Device's VLAN ID from where you want to create the Telnet session. If not configured, the default port is 23. |

Command Mode

Enable

Example

Invoking a Telnet session:

```
# telnet 10.4.4.25
```

8.11 traceroute

This command performs a traceroute and displays the route (path) and packet transit delays across an IP network, for diagnostic purposes.

Syntax

```
# traceroute { < Destination IP Address | Hostname > } [ interface
name < Interface Name > | vlan < VLAN ID > < Source IP Address > ]
```

| Arguments | Description |
|------------------------------------|--|
| Destination IP Address or Hostname | The IP address or hostname to which the trace is sent. |
| interface name | Name of the interface. |
| vlan | Defines the VLAN ID. |

Note

- Supports both IPv4 and IPv6 addresses.
- In IPv4, it supports hostname resolution as well.
- Sends three requests to each hop on the way to the destination.

Command Mode

Enable

Example

Examples of using this command:

- IPv6:

```
# traceroute ipv6 2014:6666::dddd
1 2014:7777::aa55 (2014:7777::aa55) 2.421 ms 2.022
ms 2.155 ms
2 2014:6666::dddd (2014:6666::dddd) 2.633 ms 2.481
ms 2.568 ms
Traceroute: Destination reached
```

- IPv4:

```
# traceroute 10.3.0.2
1 1 (10.4.0.1) 2.037 ms 3.665 ms 1.267 ms
2 1 (10.3.0.2) 1.068 ms 0.796 ms 1.070 ms
Traceroute: Destination reached
```

8.12 usb

This command allows maintenance on USB sticks plugged into the device.

Syntax

```
# usb list | remove
```

| Argument | Description |
|----------|---|
| list | Displays files located on the USB. |
| remove | Safely removes a USB stick that is plugged into the device. |

Command Mode

Enable

Note

Applicable only to devices providing USB port interfaces.

8.13 write

This command saves the device's current configuration to flash memory.

Syntax

```
# write
```

Command Mode

Enable

Note

The command does not reset the device. For parameters that require a reset for their settings to take effect, use the `reload now` command *instead of* the `write` command or use it *after* the `write` command.

Related Commands

```
reload now
write factory
```

Example

Saving the configuration to flash memory:

```
# write
Writing configuration...done
```

8.14 write factory

This command restores the device's configuration to factory defaults.

Syntax

```
# write factory
```

Note

When this command is run, Auxiliary files are also erased.

Command Mode

Enable

Related Commands

```
# write
```


8.15 write-and-backup

This command saves the device's configuration file to flash memory and uploads it to a specified destination. The feature provides a method to back up your saved configuration.

Syntax

```
# write-and-backup to < URL with Configuration File Name>
# write-and-backup to usb:///< Configuration File Name >
```

| Argument | Description |
|----------|---|
| to | <p>Specifies the destination to which to send the configuration file:</p> <ul style="list-style-type: none"> ▪ URL path of a remote server (TFTP or HTTP/S) ▪ USB storage stick attached to the device <p>The device first saves the configuration file to flash memory and then sends the file to the defined destination.</p> |

Command Mode

Enable

Note

- The USB option applies only to Mediant 5xx and Mediant 8xx.
- The configuration of the backed-up file is based only on CLI commands.

Related Commands

`write`

Example

- Saving a device's configuration to flash memory and sends it to a HTTP remote server:


```
# write-and-backup to http://www.example.com/configuration.txt
```
- Saving a device's configuration to flash memory and sends it to the plugged-in USB stick:


```
# write-and-backup to usb:///configuration.txt
```

International Headquarters

1 Hayarden Street,
Airport City
Lod 7019900, Israel
Tel: +972-3-976-4000
Fax: +972-3-976-4040

AudioCodes Inc.

27 World's Fair Drive,
Somerset, NJ 08873
Tel: +1-732-469-0880
Fax: +1-732-469-2298

Contact us: www.audiocodes.com/info

Website: www.audiocodes.com



Document #: LTRT-17922