LifeSize® Passport™ Automation Command Line Interface

Software Release v4.7.0

July 2010

Copyright Notice

©2009-2010 Logitech, and its licensors. All rights reserved.

Trademark Acknowledgments

Patent Notice

For patents covering LifeSize® products, refer to http://www.lifesize.com/support/legal.

Contacting Technical Services

If you have questions, concerns, or need assistance, contact your LifeSize Partner.

Providing Customer Feedback

LifeSize Communications welcomes your comments regarding our products and services. If you have feedback about this or any LifeSize product, please send it to feedback@lifesize.com. Refer to http://www.lifesize.com/support for additional ways to contact LifeSize Technical Services.

Revisions in this Release

Software release v4.7.0 for LifeSize Passport includes new, revised, and deprecated commands, objects, targets, and arguments in the LifeSize Passport Automation Command Line Interface (CLI).

New Objects and Targets

The following table identifies new objects and targets available in software release v4.7.0 for LifeSize Passport. For more information, including a list of arguments and examples, refer to the object and target descriptions in the applicable command verb chapter. Numbers that appear in parentheses following a description are for internal tracking purposes only.

New Objects and Targets

Applicable Verbs	Object	Target	Description
get set	audio	active-mic-to-lineout	New target that directs the active microphone signal to line out. (END-14362)
get set	audio	audio-output	New target that allows you to choose audio out to the display or line out for audio calls. (END-12349)
get set	audio	txgain	New target that allows you to set the gain for transmitted audio. (END-14993)
get set	audio	video-output	New target that allows you to choose audio out to the display or line out for video calls. (END-12349)
get set	call	auto-multiway	New target that allows you to control the auto answer multiway option (END-14321)
get set	call	termination-time	New target that allows you to set the number of seconds to wait before terminating a call to a busy or invalid number.
get set	call	total-bw	New target that allows you to set an upper limit for incoming and outgoing bandwidth for all legs of a call. (END-15659)

New Objects and Targets (Continued)

Applicable Verbs	Object	Target	Description
get set	camera	active	New target that allows you to set or get the active camera.
get set	camera	hflip	New target that allows you to horizontally flip the video image. (END-14584)
get set	directory	meeting	New target that allows you to access the meeting directory. (END-15101)
get set	h323	gkauth	New target to enable or disable gatekeeper authentication. (END-15661)
get set	h323	gkid	New target to enter gatekeeper authentication username. (END-15661)
get set	h323	gkpass	New target to enter gatekeeper authentication password. (END-15661)
get set	serial	port3	New object and target that adds USB support.
get set	skype	allow-from	New target that allows you to restrict from whom you receive Skype calls. (END-13031)
get set	skype	auto-logout	New target that allows you to set a time after which you are automatically logged out of the Skype server. (END-13031)
get set	system	admcontrol	New target to enable or disable admission control. (END-10327)
get set	system	corporate-dir-access	New target to hide or show the corporate directory. (END-14232)

New Objects and Targets (Continued)

Applicable Verbs	Object	Target	Description
get set	system	mcu	New target to enable or disable multiway call functionality. (END-14321)
get set	system	meetings-dir-access	New target to hide or show the meetings directory. (END-14232)
get set	system	network-presentation	New target to enable or disable network presentation with LifeSize Virtual Link. (END-14818)
get set	video	streaming	New target to configure streaming and recording. (END-15170, END-14904)
control	record	start	New target to start a recording. (END-15170)
control	record	stop	New target to stop recording. (END-15170)

Enhancements to Existing Objects or Targets

The following table identifies enhancements to existing objects or targets in this release. For more information, refer to the object and target descriptions in the applicable verb chapter. Numbers that appear in parentheses following a description are for internal tracking purposes only.

Enhancements to Existing Objects or Targets

Command	Description
control call dial	Updated protocol arguments to add ip and rtsp. (END-15298)
get/set directory ldap	Added the o argument for specifying the OU value. (END-14794)
get/set audio codecs	Added Skype SILK audio codecs as options. (END-14583)

Deprecated Commands

The following commands are deprecated in this release.

Deprecated Commands

Command	Description
get network	This command has been superseded by the `get network ipv4` and `get network ipv6` commands.
get video h241-mbps set video h241-mbps	The H.241 MaxStaticMBPS option has been deprecated from this release.
get system isdn set system isdn	This LifeSize Networker related target has been deprecated from this release.

Introduction

LifeSize Passport Automation Command Line Interface (CLI) provides a command line-based entry point for automating access and control of LifeSize Passport systems. The CLI allows you to:

- Retrieve configuration information about your LifeSize Passport system. For example, you can get the system version number or the camera's type.

- Apply new preferences to the system configuration. For example, you can set the speaker volume or the fadeout timer.

- Show the status of calls in the system. For example, you can show the active call or statistics for previous calls.

- Control aspects of the system. For example, you can emulate remote control functionality.

This document contains information about using the CLI commands, their output, and generated return codes.

Notational Conventions

The following conventions are used in this document.

Convention	Description

`monotype` font	Monotype font reflects commands and the resulting output. Constant input appears in **Bold**, for example:
	
	`get system uptime`
	
	Variable input appears in ***Bold Italic***, for example:
	
	`set audio mics off`
	
	Constant output appears in plain `monotype`, for example:
	
	`ok,00`
	
	Variable output appears in `monotype` *Italic*, for example:
	
	`get audio video-output`
	
	`phone`

angle brackets `<>`	Required parameters are enclosed in angle brackets, for example: `<parameter>`

square brackets `[]`	Optional parameters are enclosed in square brackets, for example:
	
	`[parameter]`
	
	Similarly, optional arguments are enclosed in square brackets, for example:
	
	`[-p]`

Convention	Description

curly brackets { }	Parameters whose values are restricted are enclosed in curly brackets with discrete values separated by a pipe (|) symbol. The following example restricts the values to val1, val2, or val3:
	
	`<{val1|val2|val3}>`
	
	Parameters whose values are restricted to a range of values are enclosed in curly brackets and separated by a pair of periods (..). The following example restricts values to integers between 0 and 100, inclusive:
	
	`[{0..100}]`
	
	The following example restricts values to integers between -30 and 30, inclusive:
	
	`[-p {-30..30}]`
	
	The following example restricts values to floating point numbers between -30.0 and 30.0 inclusive:
	
	`[-p {-30.0..30.0}]`

Fundamentals of the Command Line Interface

This chapter describes the fundamental concepts of the CLI, such as accessing the CLI, help and default output modes, command line syntax, and standard output format.

Accessing the Command Line Interface

The CLI is available through an ssh or telnet connection to your LifeSize video communications system as the **auto** user (default password **lifesize**).

Help Mode

The CLI has two modes of operation: normal mode and help mode. By default, the CLI starts in help mode.

In help mode, full command help is available, as is abbreviation support. The help mode setting exists only for the duration of the current instance. It is not shared between multiple instances. Help mode is enabled by default but can be controlled through the **set help-mode** command. Because help mode also enables command abbreviations, LifeSize recommends that you run automated scripts or programs with the help mode set to **off** to prevent using abbreviations in these types of situations. Refer to "help-mode" on page 90 for details about using this command.

All commands provide basic usage information when you specify the **-h** option to the command at any point in the argument list. **Help** followed by a verb produces the list of targets for that verb. Additionally, the argument **errors** returns a list of error message codes and their meanings. In normal mode, entering **help** produces an unsupported verb error.

Examples:

> **help**
>
> error,09
>
> **set help-mode on**
>
> ok,00

```
help
Possible verbs:
control
get
history
set
status

ok,00


help set
Possible completions:
set admin password
set audio mics
set call auto-answer
set volume dtmf
set volume ring-tone

ok,00
```

Note: The previous examples show only a subset of all possible completions.

```
help errors -V
Code    Description
00      Success
01      No Memory
02      File Error
03      Invalid Instance
04      Invalid Parameter
05      Argument is not repeatable
06      Invalid Selection Parameter Value
07      Missing Argument
08      Extra Arguments on Command Line
09      Invalid Command
0a      Ambiguous Command
0b      Conflicting Parameter
0c      Operational Error
0d      No Data Available
0e      Not In Call
0f      Interrupted
10      Ambiguous Selection
11      No Matching Entries
12      Not Supported


ok,00
```

Default Output Mode

The CLI supports a default output mode option. Like help mode, it exists only for the duration of the CLI instance and is not shared between instances. By default, terse output mode is enabled. You can change this mode using the `set verbose-mode on` command. This is equivalent to specifying the `-v` option to each command entered.

>   ***Note:***   Asynchronous status messages are always printed in terse mode using the
>   default delimiter, regardless of the current state of verbose mode or any
>   delimiter option used on the command that caused the asynchronous message
>   to occur.

Command Line Arguments

You can invoke a single command by specifying that command on the command line, for example:

        `ssh auto@lifesize get camera position`

In this example, the return code of the ssh command is the result code from the single command executed.

Command Line History and Recall

The CLI also supports command line history, editing, and recall through the editline library. These features operate in a similar manner to the GNU bash shell, including support for `!n`, `!!` and Emacs editing modes. History is limited to the last 100 commands.

Here Documents

The CLI supports a scripting feature known as a here document. When used in the CLI, a here document is a block of data that can be fed to certain commands that accept several lines of input (for example, uploading images or files to the system). Descriptions and examples in this manual indicate support for here documents when available for a command. Following is the syntax for specifying a here document in a CLI command:

command << TOKEN

input_associated_with_command

TOKEN

where the here document consists of all text between the *TOKEN* document start symbol and the *TOKEN* document end symbol. The start symbol and end symbol must be identical. The input does not include the new line after the start symbol, but does include the new line

immediately before the end symbol. The end symbol must start in the first column of a new line to be recognized. Here documents are generally used for sending scripts to the CLI through an SSH session. For example:

Manually enter an ssh key using a here document:

```
set ssh keys -i << EOF
ssh-rsa key_string user@lifesize.com
ssh-rsa key2_string user2@lifesize.com
EOF
```

Upload a background image using a here document:

```
set video background image << EOF
<base 64 encoded data stream>
EOF
```

Command Syntax

In general, the syntax is relatively rigid to ensure consistency across all commands that the CLI supports.

The general syntax of a command is *<verb> <object> <target>* [*options*], where:

<verb> defines the operation to perform.

<object> defines the subsystem on which the operation should be performed.

<target> identifies the specific parameter within the object.

[*options*] specifies arguments that may be passed in the command.

Note: Unless otherwise indicated, when specifying an argument that includes a text string with a space in the string, enclose the text in double quotes (for example, "QRB Meeting").

Command Verbs

The CLI verbs are `get`, `set`, `control`, `status`, `exit`, `help`, and `history`.

If help mode is enabled, help is available for the verbs, objects, and targets. In this context, a complete command is defined as a verb followed by an object and complete target specification. (For two word targets, you must specify both to complete the command). If you specify an incomplete command, all possible completions for that command root are displayed in alphabetical order. Additionally, the command processor allows abbreviations of command targets and verbs to simplify usage and to allow for more descriptive targets.

LifeSize recommends you do not use abbreviations in shell scripts, because future releases may make the abbreviation ambiguous. To prevent such use, abbreviations are disabled when help mode is off.

Command Verb	Description
`get`	The `get` verb retrieves preference configuration information from the system (for example, displaying the current audio levels).
`set`	The `set` verb applies new preferences to the system configuration (for example, changing the speaker volume).
`control`	The `control` verb initiates an action on the system (for example, placing a call).
`status`	The `status` verb retrieves system status information (for example, call information).

`exit`	The `exit` verb exits the shell prior to the end of input. The exit verb has no arguments. *Example*:
	
	`exit`
	
	`ok,00`
	
	You can also exit the shell by entering the end-of-file character (generally **^D**).

`help`	The `help` verb is available only in help mode. It lists the verbs available in the shell but does not list the individual targets for those verbs. `Help` followed by a verb produces the list of targets for that verb (as if just the verb had been entered on the command line).

`history`	The `history` verb lists the saved history of commands up to 100 lines. Blank and commented lines are not included. To limit the number of lines displayed to fewer than 100, type the verb followed by the number of lines to display. To execute a command from the history list, type *!x*, where *x* is the number of the command. For example, if the `history` verb displayed the following history of commands:
	
	`history`
	
	`1,control remote back back`
	
	`2,get camera position`
	
	`3,control call dial redial:1`
	
	`ok,00`
	
	then, *!2* would execute command 2 (`get camera position`). The history persists across shell invocations.

Standard Options

All of the command verbs support a small set of standard command line options to provide a basic level of consistency.

Provide Help: -h

All commands provide basic usage information for interactive users. Specify the **-h** option in the command at any point in the argument list. When you specify **-h** at any level other than that of a completed command, a list of all possible completions appears (**-h** is ignored in this case). Command help is available only when help mode is enabled. For example:

```
get system model -h
Usage: get system model [-?] [-D c] [-V] [-h]
-?          Display the column headers, even in terse mode
-D c        Specify an alternate delimiter character in
            terse mode (default is ',')
-V          Enable verbose output mode
-h          Produce this message

ok,00
```

Enable Verbose Output: -V

By default, command output appears in terse format suitable for processing by scripts. If you specify **-V**, output appears in a tabular format with headers describing each column. A minimum of two spaces separate each column value. This format is suitable for human parsing and for use during prototyping. The order of the columns presented in verbose and terse modes is the same, so you can rely on the output in verbose mode to guide column selection in terse mode. To enable verbose permanently, set verbose-mode to **on**.

Set the Terse Mode Column Delimiter: -D <c>

The default column delimiter in terse mode is the comma (',') character. Use the **-D** option to change the delimiter to any single character other than space (ASCII 0x20) or newline (ASCII 0x0a). The first character of the argument to **-D** is the new delimiter character. When outputting data in terse mode, any occurrence of the delimiter character in the output is replaced with the space character. The **-D** option and the **-V** option (enabling verbose mode as a default) are mutually exclusive. If both are specified, **-D** is ignored. For example:

```
get system model -D |
LifeSize|Passport

ok|00
```

Standard Output Format

All of the internal commands produce output in a specific format, based on the default output mode or the presence of the **-v** option.

Terse Mode Output

Terse mode is the default output mode. It is designed to be easily parsed by shell scripts and automated programs. The general format of the output is rows of comma-separated text. To change the separation character, specify the **-D** option. The completion code for the command is also sent to the output stream. For example:

```
get network ipv4
static,10.10.100.5,255.255.255.0,10.10.100.1,00:13:fa:00:24:a1,
    jsmith-ls

ok,00

get unknown-target

error,09
```

To allow differentiation between command output and the completion code output, a single newline is always inserted between the last line of command output and the completion code. Command output is not allowed to contain any blank lines. The completion code is printed as *<status>*,*<code>* where status is either `ok` or `error` and code is a two digit hexadecimal number. A code value of `00` indicates success of the command. Any other value indicates an error condition.

Verbose Mode Output

Verbose mode is enabled by specifying the **-v** option to a command. It may also be enabled globally by setting verbose mode to **on**. Verbose mode is designed for human parsing and is formatted in a tabular style. Verbose mode is not intended to be parsed by automated scripts. For example:

```
get network ipv4 –V
Type    IP Address    Network Mask   Gateway       MAC Address
static  10.10.100.5  255.255.255.0  10.10.100.1   00:13:fa:00:24:a1

ok

get unknown-target

error 09 Target not recognized
```

Show Column Headings in Terse Mode: -?

To show column headings from verbose mode while in terse mode, use the `-?` option in a command. In this mode, the column headings from verbose mode appear on the first line of output separated by commas, followed by terse mode output on the next line. For example:

```
get system model -?
OEM,Model
LifeSize,Passport

ok,00
```

Standard Return Codes

All CLI commands return a standard error code on completion. You can access the following table of return codes using the `help errors` command.

Return Code		Mnemonic	Description
Dec	**Hex**		
0	00	Ok	The command completed successfully.
1	01	NoMemory	The command failed due to a loss of memory.
2	02	IOError	The command failed due to a file read/write/open error.
3	03	InvalidInstance	The command failed due to data corruption.
4	04	InvalidParameter	An incorrect option or argument was specified on the command line.
5	05	Repeated	A non-repeatable option or argument was repeated.
6	06	NotInList	The specified option or argument value was not in the selection list.
7	07	Missing	A required option or argument was not specified.
8	08	TooMany	Too many arguments were specified.
9	09	InvalidCommand	The command entered was not found.
10	0a	AmbiguousCommand	The command entered is ambiguous.
11	0b	ParameterConflict	Two or more mutually exclusive options were specified.
12	0c	OperationalError	The command failed for unspecified reasons.

Return Code		Mnemonic	Description
Dec	Hex		
13	0d	NoData	No data is available for this operation (no active calls) or the command timed out.
14	0e	NotInCall	The command requires an active call for operation.
15	0f	Interrupted	The command was interrupted.
16	10	Ambiguous	The directory selection is ambiguous (matches multiple entries).
17	11	NoMatch	The directory selection does not match any entries.
18	12	NotSupported	The far end of the call does not support presentations.

Generating the Command Listing

To generate a complete list of the commands available in the CLI shell, execute the following:

```
% set help-mode on

ok,00
% help
control
get
exit
help
history
set
status

ok,00
% control
<control command list>

ok,00
% get
<get command list>
...
```

The result is a list of the supported commands available in the CLI with the exception of the help, history, and exit top level commands.

get and set Verbs: Objects and Targets

This chapter identifies objects and targets that apply to the **get** and **set** verbs. Most of the objects and targets apply to both verbs. Where only one of the verbs applies, the description and examples specify the verb.

admin

The **admin** object controls configuration of administrator functions in the interface. This object applies to the **set** verb. The following target applies to the **admin** object.

password

The **password** target sets the password for access to the administrator preferences. This target applies to the **set** verb.

Arguments:

`<value>`	The new administrator password. The password can be an empty string, the numbers 0-9 and/or the symbols * and #. The password is silently truncated to 16 characters.

Examples:

```
set admin password 12345*#

ok,00

set admin password -V abcdef

error  04  Invalid Parameter
```

audio

The following targets apply to the **audio** object.

active-mic

When used with the **get** verb, the **active-mic** target shows the option set for the active microphone preference. To view which input device is the active microphone, use the **get audio current-mic** command.

When used with the **set** verb, this target specifies the option to use for active microphone.

get Arguments:

    None

get Examples:

```
get audio active-mic
auto

ok,00


get audio active-mic -V
Active
hd0

ok
```

set Arguments:

`<{auto|micin|hd0}>`	Specify the active microphone. The `micin` argument specifies a microphone connected to the microphone input on the codec. Specify `hd0` for LifeSize Focus as the active microphone. The `auto` argument specifies a default order of selection based on the inputs available on the system: `micin`, `hd0`.

set Examples:

```
set audio active-mic auto

ok,00
```

active-mic-to-lineout

When used with the **get** verb, the `active-mic-to-lineout` target shows whether the audio signal from the active microphone is sent to the line out (`enabled`) or not (`disabled`). When used with the **set** verb, this target controls whether the audio signal from the active microphone is sent to the line out.

get Arguments:

   None

get Examples:

```
get audio active-mic-to-lineout
disabled

ok,00

get audio active-mic-to-lineout -V
State
disabled

ok
```

set Arguments:

`<{enabled|disabled}>`	Specify `enabled` to send the audio signal from the active microphone to the line out.

set Examples:

```
set audio active-mic-to-lineout enabled

ok,00
```

audio-output

When used with the **get** verb, the audio-output target shows the output destination that is set for audio output when placing a voice call. When used with the **set** verb, this target specifies the output destination for audio from a voice call to the line out or the HD out on the system.

get Arguments:

    None

get Examples:

```
get audio audio-output
hdmi

ok,00

get audio audio-output -V
Destination
room

ok
```

set Arguments:

`<{room|hdmi}>`	Specify the output port for voice call audio. Choose `room` to send the audio to the line out port. Choose `hdmi` to send the audio to the HD out port.

set Examples:

```
set audio audio-output room

ok,00
```

codecs

When used with the **get** verb, the **codecs** target retrieves the codec priority list. This list is in the order in which audio codecs are negotiated when connecting to other systems. When used with the **set** verb, this target changes the negotiating order of the audio codecs. The list is ordered from highest priority to lowest priority. For greatest compatibility, list all available codecs. You can list each codec only once.

get Arguments:

    None

audio

get Examples:

```
get audio codecs
aac-lc g.722.1c.48 g.722.1c.32 g.722.1c.24 silk.24 silk.16
   silk.12 silk.8 g.722 g.729 g.728 g.711.u g.711.a

ok,00

get audio codecs -V
Codec Order
aac-lc g.722.1c.48 g.722.1c.32 g.722.1c.24 silk.24 silk.16
   silk.12 silk.8 g.722 g.729 g.728 g.711.u g.711.a

ok
```

Following are the available codecs:

Codec	CLI Name
AAC Low Complexity	aac-lc
Polycom® Siren14™ (48 kb/s)	g.722.1c.48
Polycom® Siren14™ (32 kb/s)	g.722.1c.32
Polycom® Siren14™ (24 kb/s)	g.722.1c.24
Skype SILK 24	silk.24
Skype SILK 16	silk.16
Skype SILK 12	silk.12
Skype SILK 8	silk.8
G.722	g.722
G.729	g.729
G.728	g.728
G.711 μ-Law	g.711.u
G.711 A-Law	g.711.a

set Arguments:

`<{aac-lc|g.722.1c.24|`
`g.722.1c.32|g.722.1c.48|`
`silk.24|silk.16|silk.12|`
`silk.8|g.722|g.728|`
`g.729|g.711.u|g.711.a}>`
	Specify the order of the audio codecs to use. List each codec only once. LifeSize recommends that you list each codec on the command line for greatest compatibility.

set Example:

set audio codecs *aac-lc g.722 g.722.1c.48 g.722.1c.32 g.722.1c.24*
    silk.24 silk.16 silk.12 silk.8 g.711.a g.711.u g.728 g.729

ok,00

current-mic

The `current-mic` target shows the input currently used as the active microphone for audio input during calls. This information appears in the System Information page in the user interface as the value of the **Active Microphone** field. This target applies to the `get` verb.

Arguments:

   None

Examples:

get audio current-mic
hd0

ok,00
get audio current-mic -V
Value
hd0

ok

audio

eq

When used with the **get** verb, the **eq** target retrieves the bass and treble equalization settings. When used with the **set** verb, this target sets the bass and treble equalization parameters.

get Arguments:

   None

get Examples:

   get audio eq
   0,-3

   ok,00

   get audio eq -V
   Bass   Treble
   4   -5

   ok

set Arguments:

[-b {-10..10}]	Specify the bass equalization value.
[-t {-10..10}]	Specify the treble equalization value.

set Examples:

   set audio eq -b *-5* **-t** *6*

   ok,00

gain

When used with the **get** verb, the **gain** target retrieves the current setting for the active microphone volume. When used with the **set** verb, this target specifies the setting for the active microphone volume on a scale of 0 to 20.

get Arguments:

    None

get Examples:

    get audio gain
    5

    ok,00

    get audio gain -V
    Gain
    5

    ok

set Arguments:

`<{0..20}>`	Specify the gain factor for the microphone input. Use larger numbers for more gain.

set Examples:

    set audio gain *8*

    ok,00

audio

levels

The **levels** target retrieves the volume levels for audio inputs that have audio meters (active microphone) and any connected calls. This target applies to the **get** verb. The values that return include the following:

- Session—the name of the audio input or, in an active call, the call handle. The input names that appear in the output are the same as the arguments for the **set audio active-mic** command except the argument auto does not appear in the output. CurrentMic—identifies whether or not the input is currently used as the active microphone for audio input during calls. This information also appears in the System Information page in the user interface as the value of the **Active Microphone** field and as the output of the **get audio current-mic** command.

- Power—the volume expressed as decibels (dB) below digital full scale with 0 as the maximum value (-140 to 0)

- PowerAvg—the power averaged over 750 milliseconds and expressed as dB below full digital scale (-140 to 0)

- PowerPCT—the power averaged over 500 milliseconds and converted to a range (0 to 100)

get Arguments:

   None

get Examples:

   LifeSize Passport with LifeSize Focus as the active microphone:

   get audio levels
   hd0,yes,-84,-84,0

   ok,00

   get audio levels -V

Session	CurrentMic	Power	PowerAvg	PowerPCT
hd0	*yes*	*-84*	*-84*	*0*

   ok

LifeSize Passport Automation Command Line Interface

mute

When used with the **get** verb, the **mute** target retrieves the current setting of the local audio mute function. When used with **set** verb, this target controls whether the local audio inputs are muted.

get Arguments:

> None

get Examples:

> **get audio mute**
> off
>
> ok,00
>
> **get audio mute -V**
> State
> *on*
>
> ok

set Arguments:

`<{on|off}>`	Mute or unmute the local audio inputs.

set Examples:

> **set audio mute *on***
>
> ok,00

ringer

When used with the **get** verb, the **ringer** target returns *enabled* or *disabled* to indicate whether the system ringer is activated. When used with **set** verb, this target enables or disables LifeSize Passport's built-in ringer.

get Arguments:

> None

get Examples:

> **get audio ringer**
> *enabled*
>
> ok,00

audio

```
get audio ringer -V
Ringer
disabled

ok
```

set Arguments:

`<{enabled|disabled}>`	Specify `enabled` to activate LifeSize Passport's built-in ringer. Specify `disabled` to deactivate it.

set Examples:

```
set audio ringer enabled

ok,00
```

ringer-type

When used with the **get** verb, the **ringer-type** target shows the ringer tone setting (1-5) for LifeSize Passport. When used with the **set** verb, this target sets the ring tone (1-5) for the ringer.

get Arguments:

> None

get Examples:

```
get audio ringer-type
3

ok,00
```

```
get audio ringer-type -V
Ringer Type
3

ok
```

set Arguments:

`<{1..5}>`	Specify a ringer type, 1 through 5.

set Examples:

```
set audio ringer-type 4

ok,00
```

ringer-volume

When used with the **get** verb, the **ringer-volume** returns the ringer volume. When used with **set** verb, this target sets the ringer volume (0-10).

get Arguments:

　　None

get Examples:

```
get audio ringer-volume
5

ok,00

get audio ringer-volume -V
Ringer Volume
5

ok
```

set Arguments:

`<{0...10}>`	Specify a ringer volume between 0 and 10.

set Examples:

```
set audio ringer-volume 4

ok,00
```

test-tone

The **test-tone** target sends a test tone to the audio output on the codec. Use this target to verify the speakers are connected correctly when installing the system. The **test-tone** target applies to the **set** verb.

Arguments:

`[-l]`	Send the test tone to the left channel speaker output.
`[-r]`	Send the test tone to the right channel speaker output.
`[-o]`	Turn off the test tone.
`[-s n]`	Send the test tone to the left and right outputs in order for five seconds for *n* cycles. The command does not return output until all cycles complete.

Note: Only one option can be specified at a time.

Examples:

set audio test-tone -r

ok,00

set audio test-tone -o

ok,00

When everything is configured, verify with:

set audio test-tone -s *1*

ok,00

txgain

When used with the **get** verb, the **txgain** target retrieves the current setting for the transmitted audio volume. When used with the **set** verb, this target specifies the setting for the transmitted audio volume on a scale of 0 to 7.

get Arguments:

    None

get Examples:

    get audio txgain
    0

    ok,00

    get audio txgain -V
    Gain
    3

    ok

set Arguments:

`<{0..7}>`	Specify the gain factor for the transmitted audio. Use larger numbers for more gain.

set Examples:

    set audio txgain *3*

    ok,00

audio

video-output

When used with the **get** verb, the **video-output** target retrieves the current output destination for audio received during a video call. When used with the **set** verb, this target controls whether the video call audio output is sent to line out (room) or the primary display (hdmi).

get Arguments:

    None

get Examples:

```
get audio video-output
hdmi

ok,00

get audio video-output -V
Destination
room

ok
```

set Arguments:

<{room|hdmi}>	Choose line out (room), or HD out (hdmi) as the audio output device for video calls.

set Examples:

```
set audio video-output hmdi

ok,00
```

call

The following targets apply to the **call** properties object.

auto-answer

When used with the **get** verb, the **auto-answer** target retrieves the current value of the auto answer setting. When used with the **set** verb, this target controls whether or not the system automatically answers an incoming call.

get Arguments:

>   None

get Examples:

>   **get call auto-answer**
>   *on*
>
>   ok,00
>
>   **get call auto-answer -V**
>   State
>   *off*
>
>   ok

set Arguments:

<{on|off}>	Enable or disable auto answer for an incoming call.

set Examples:

>   **set call auto-answer *on***
>
>   ok,00

call

auto-bandwidth

When used with the **get** verb, the **auto-bandwidth** target retrieves the current setting for automatic bandwidth negotiation when placing or receiving calls. When used with the **set** verb, this target enables or disables automatic bandwidth negotiation when placing and answering calls.

get Arguments:

    None

get Examples:

    get call auto-bandwidth
    on

    ok,00

    get call auto-bandwidth -V
    Bandwidth Negotiation
    off

    ok

set Arguments:

<{on|off}>	Enable or disable automatic bandwidth negotiation.

set Examples:

    set call auto-bandwidth *on*

    ok,00

auto-multiway

When used with the **get** verb, the **auto-multiway** target retrieves the current value of the auto answer multiway call preference. When used with the **set** verb, the **auto-multiway** target controls whether the system automatically answers incoming multiway calls after the first call has connected.

get Arguments:

    None

get Examples:

    get call auto-multiway
    on

    ok,00

    get call auto-multiway -V
    State
    off

    ok

set Arguments:

<{on|off}>	Enable or disable auto answer of multiway calls.

set Examples:

    set call auto-multiway *off*

    ok,00

auto-mute

When used with the **get** verb, the **auto-mute** target retrieves the current value of the auto answer mute preference, which indicates whether or not the audio input devices are muted when the system automatically answers a call. When used with the **set** verb, this target controls whether or not the system automatically mutes the microphone inputs when it automatically answers the call.

get Arguments:

    None

get Examples:

```
get call auto-mute
on

ok,00

get call auto-mute -V
State
off

ok
```

set Arguments:

<{on|off}>	Enable or disable muting of the microphone when answering a call.

set Examples:

```
set call auto-mute off

ok,00
```

dial-mode

When used with the **get** verb, the **dial-mode** target retrieves the current settings for the voice and video dialing preferences. When used with the **set** verb, this target configures the default audio and video dialing modes.

get Arguments:

   None

get Examples:

   get call dial-mode
   ip,voip

   ok,00

   get call dial-mode -V
   Video Dial Mode   Audio Dial Mode
   ip   *voip*

   ok

set Arguments:

`[-a {voip|` `isdn|skype}]`	Set the default voice call dialing mode to VoIP, ISDN, or Skype.
`[-v {auto|ip|` `isdn|skype}]`	Set the default video call dialing mode to auto, IP, ISDN, or Skype. Auto attempts to determine the correct dialing mode from the dialed digits.

set Examples:

   set call dial-mode -a *voip* -v *ip*

   ok,00

   set call dial-mode -v *auto*

   ok,00

call

max-redial-entries

When used with the **get** verb, the **max-redial-entries** target retrieves the value of the setting for the maximum number of redial entries that may be stored in the directory. When used with the **set** verb, this target controls the number of redial entries that can be stored in the redial list.

get Arguments:

>   None

get Examples:

>   **get call max-redial-entries**
>   *11*
>
>   ok,00
>
>   **get call max-redial-entries -V**
>   Number
>   *9*
>
>   ok

set Arguments:

`<{0..15}>`	Specify the maximum number of entries in the redial list.

set Examples:

>   **set call max-redial-entries** *11*
>
>   ok,00

max-speed

When used with the **get** verb, the **max-speed** target retrieves the current setting for the maximum incoming and outgoing bandwidth usable by a call. Speed is reported in kilobits per second (kb/s). When used with the **set** verb, this target controls the current settings for the maximum incoming and outgoing bandwidth usable by a call.

get Arguments:

   None

get Examples:

   get call max-speed
   1024,768

   ok,00

   get call max-speed -V
   Incoming   Outgoing
   512   1024

   ok

set Arguments:

`[-i {auto|128|192|256|320|384|`
`512|640|768|896|1024|1152|1280|`
`1472|1728|1920|2000}]`
	Specify the maximum incoming bandwidth in kb/s to be used by a call.

`[-o {auto|128|192|256|320|384|`
`512|640|768|896|1024|1152|1280|`
`1472|1728|1920|2000}]`
	Specify the maximum outgoing bandwidth in kb/s to be used by a call.

set Examples:

   set call max-speed -i *512* **-o** *auto*

   ok,00

max-time

When used with the **get** verb, the **max-time** target retrieves the maximum time that a call can be active. When used with the **set** verb, this target controls the maximum time that a call can be active.

get Arguments:

>   None

get Examples:

>   **get call max-time**
>   *unlimited*
>
>   ok,00
>
>   **get call max-time -V**
>   Time In Hours
>   *4*
>
>   ok

set Arguments:

`<{30-minutes|1|2|4|6|8|10|12 |14|16|18|20|22|24|unlimited}>`	Specify the maximum time in hours that a call can be active.

set Examples:

>   **set call max-time *unlimited***
>
>   ok,00
>
>   **set call max-time *4***
>
>   ok,00

termination-time

When used with the **get** verb, the **termination-time** target shows the number of seconds a LifeSize system waits before terminating a call to a busy or invalid number. When used with the **set** verb, this target specifies the number of seconds the system waits before terminating a call to a busy or invalid number. The default is 30 seconds.

get Arguments:

>   None

get Examples:

>   **get call termination-time**
>   *30*
>
>   ok,00
>
>   **get call termination-time -V**
>   Seconds
>   *30*
>
>   ok

set Arguments:

`<{0..300}>`	Specify the number of seconds the system waits before terminating a call to a busy or invalid number.

set Examples:

>   **set call termination-time 60**
>
>   ok, 00

call

total-bw

When used with the **get** verb, the **total-bw** target shows the incoming and outgoing bandwidth limit for all calls, if set. When used with the **set** verb, this target specifies the incoming and outgoing bandwidth limit for all calls.

get Arguments:

   None

get Examples:

   get call total-bw
   1152,1152

   ok,00

   get call total-bw -V
   incoming   Outgoing
   1152   1152

   ok

set Arguments:

`[-i {auto|128|192|256|320|384|`
`512|640|768|896|1024|1152|1280|`
`1472|1728|1920|2000}]`
	Specify the maximum total incoming bandwidth in kb/s for all calls.

`[-o {auto|128|192|256|320|384|`
`512|640|768|896|1024|1152|1280|`
`1472|1728|1920|2000}]`
	Specify the maximum total outgoing bandwidth in kb/s for all calls.

set Examples:

   set call total-bw -i 1152 -o 1152

   ok,00

camera

The following targets apply to the **camera** object.

active

When used with the **get** verb, the **active** target retrieves the current active high definition camera, which is always camera 1 on LifeSize Passport. When used with the **set** verb, this target sets the active camera.

get Arguments:

   None

get Examples:

```
get camera active
1

ok,00

get camera active -V
Active Camera
1

ok
```

set Arguments:

<{1|a}>	Specify the active camera. The a option is the currently active camera.

set Examples:

```
set camera active 1

ok,00
```

camera

anti-flicker

When used with the **get** verb, the **anti-flicker** target retrieves the current anti-flicker setting for the cameras. When used with the **set** verb, this target controls the flicker avoidance setting of the cameras. If the local video image flickers, try the various arguments to see if the flicker goes away. Flicker is usually caused by fluorescent lighting.

get Arguments:

>   None

get Examples:

```
get camera anti-flicker
auto

ok,00

get camera anti-flicker -V
Mode
60hz

ok
```

set Arguments:

`<{auto|50hz|60hz}>`	Specify the anti-flicker setting for the cameras.

set Examples:

```
set camera anti-flicker 50hz

ok,00
```

autoexposure

When used with the **get** verb, the **autoexposure** target shows whether automatic adjustment of the camera iris to achieve the specified camera brightness setting for LifeSize Camera or LifeSize Focus connected to the codec is enabled (the default) or disabled.

When used with the **set** verb, this target disables or enables automatic adjustment of the camera iris. When autoexposure is set to *disable*, you can adjust exposure manually with the **autoexposure-gain** and **autoexposure-itime** targets to control video image brightness. Manually adjusting exposure may be useful in video conference settings where participants are backlit or where lighting sources vary.

Note: Adjusting the **HD Camera Brightness** preference in the user or web administrator interfaces has no effect when **autoexposure** is set to disable.

get Arguments:

None

get Examples:

```
get camera autoexposure
enable

ok,00

get camera autoexposure -V
State
enable

ok
```

set Arguments:

<{enable|disable}>	Enables or disables automatic adjustment of the camera iris. The default is enable.

set Examples:

```
set camera autoexposure disable

ok,00
```

camera

autoexposure-gain

When used with the `get` verb, the `autoexposure-gain` target shows the current exposure gain setting for LifeSize Camera or LifeSize Focus connected to the codec. The default is 15. The range is 0-30. When used with the `set` verb, this target controls the level of exposure gain when the `autoexposure` target is set to `disable`. Higher values for this target increase video image brightness; lower values decrease video image brightness.

Note: If `autoexposure` is set to `enable`, setting `autoexposure-gain` has no effect.

get Arguments:

None

get Examples:

```
get camera autoexposure-gain
15

ok,00

get camera autoexposure-gain -V
State
15

ok
```

set Arguments:

`<{0..30}>`	Specifies exposure gain for cameras connected to the codec. The default is 15. Higher values increase video image brightness; lower values decrease video image brightness.

set Examples:

```
set camera autoexposure-gain 20

ok,00
```

autoexposure-itime

When used with the **get** verb, the **autoexposure-itime** target shows the current setting of the exposure integration time for LifeSize Camera and LifeSize Focus connected to the codec. Exposure integration time controls the number of sensor rows in which to accumulate charge for a single pixel in the video image. The default value is 5. The range is 0-13. When used with the **set** verb, this target controls exposure integration time when the autoexposure target is set to *disable*. Higher values increase video image brightness; lower values decrease video image brightness.

Caution: Changing the default for this setting may affect the frame rate of the video. Values 0-5 produce video at 30 frames per second (f/s); values 6-9 produce video at 15 f/s; and values 10-13 produce video at 7.5 f/s. Below 30 f/s, video motion may be choppy. LifeSize recommends that you set this target to a value equal to or less than 5.

Changing the value of **autoexposure-itime** may require that you change the value of **autoexposure-gain** to achieve an acceptable level of video image brightness.

Note: If the **autoexposure** target is set to *enable*, setting **autoexposure-itime** has no effect.

get Arguments:

None

get Examples:

```
get camera autoexposure-itime
5

ok,00

get camera autoexposure-itime -V
State
5

ok
```

set Arguments:

`<{0..13}>`	Specifies exposure integration time for cameras connected to the codec. The default is 5. Higher values increase video image brightness; lower values decrease video image brightness.

camera

set Examples:

set camera autoexposure-itime *4*

ok,00

autofocus

When used with the **get** verb, the **autofocus** target shows whether automatic focus control is enabled (the default) or disabled for PTZ cameras connected to the codec. When used with the **set** verb, this target enables or disables automatic focus control. Disabling autofocus prevents the camera from automatically adjusting focus as participants move in the room or the camera position is changed. Ensure that you adjust the camera focus to the desired setting before setting this target to disable.

Note: Locking a camera automatically disables auto focus for that camera. Using this target with the **set** verb has no effect on LifeSize Focus, while the **get** verb will always return enable.

get Arguments:

None

get Examples:

get camera autofocus
enable

ok,00

get camera autofocus -V
State
enable

ok

set Arguments:

<{enable|disable}>	Enables or disables autofocus for PTZ cameras connected to the codec. The default is enable.

set Examples:

set camera autofocus *disable*

ok,00

brightness

When used with the **get** verb, the **brightness** target retrieves the current brightness adjustment value for the camera. When used with the **set** verb, this target controls the brightness value for the camera's automatic iris function. Negative numbers darken the image; positive numbers lighten it.

Note: The **set camera brightness** command has no effect on LifeSize Camera and LifeSize Focus when **get camera autoexposure** is *disable*. Use **set camera autoexposure-gain** and **set camera autoexposure-itime** to adjust the exposure in that case.

get Arguments:

   None

get Examples:

get camera brightness
-20

ok,00

get camera brightness -V
Camera 1
25

ok

set Arguments:

[-N <{1|a}>]	Optional: Specify the camera. Because only one camera is possible, camera 1, and camera a (active), are essentially the same.
<{-30..30}>	Specify the brightness adjustment value. Negative numbers darken the image; positive numbers lighten it. If you specify a negative number, include the double dash option (--) before the negative number to prevent the CLI from interpreting the negative number as an option rather than an argument. Refer to the following examples.

set Examples:

set camera brightness *3*

ok,00

camera

set camera brightness -- *-5*

ok,00

far-control

When used with the **get** verb, the **far-control** target retrieves the current state of the far control of the near camera setting. When used with the **set** verb, this target enables or disables far control of the near camera function. LifeSize Focus does not support camera control.

get Arguments:

None

get Examples:

get camera far-control
enabled

ok,00

get camera far-control -V
State
disabled

ok

set Arguments:

<{enabled|disabled}>	Specify whether far control of the near camera is enabled or disabled.

set Examples:

set camera far-control *disabled*

ok,00

far-set-preset

When used with the **get** verb, the **far-set-preset** target shows whether or not the far end of a call can set local presets. When used with the **set** verb, this target controls whether or not the far end of a call can set presets on the near PTZ camera. LifeSize Focus does not support camera control.

get Arguments:

>   None

get Examples:

>   **get camera far-set-preset**
>   *disabled*
>
>   ok,00
>
>   **get camera far-set-preset -V**
>   State
>   *enabled*
>
>   ok

set Arguments:

`<{enabled|disabled}>`	Specify whether or not the far end can set presets on the near camera.

set Examples:

>   **set camera far-set-preset *enabled***
>
>   ok,00

camera

far-use-preset

When used with the **get** verb, the **far-use-preset** target shows whether or not the far end of a call can move the near camera to local presets. When used with the **set** verb, this target controls whether or not the far end of a call can move the near PTZ camera to local presets. LifeSize Focus does not support camera control.

get Arguments:

> None

get Examples:

> **get camera far-use-preset**
> *disabled*
>
> ok,00
>
> **get camera far-use-preset -V**
> State
> *enabled*
>
> ok

set Arguments:

<{enabled|disabled}>	Specify whether the far end can move to presets.

set Examples:

> **set camera far-use-preset** *disabled*
>
> ok,00

hflip

When used with the **get** verb, the **hflip** target reveals whether the camera's image is flipped horizontally (*on*), normal (*off*), or is not set (no value). When used with the **set** verb, this target flips the camera image horizontally when set to *on*.

get Arguments:

> None

get Examples:

> **get camera hflip**
> *off*
>
> ok,00

```
get camera hflip -V
Camera 1
on

ok,00
```

set Arguments:

`[-N <{1|a}>]`	Optional: Specify the camera image to flip. Only one camera is possible.
`<{on|off}>`	Enable or disable horizontal image flipping on the specified camera.

set Examples:

```
set camera hflip on

ok,00

set camera hflip -N 1 off

ok,00
```

ir

When used with the **get** verb, the **ir** target shows whether the infra-red (IR) receiver on the camera is on or off. When used with the **set** verb, this target controls whether the IR receiver is on or off. When set to *off*, the system does not respond to commands from the LifeSize remote control.

get Arguments:

None

get Examples:

```
get camera ir
on

ok,00
```

camera

```
get camera ir -V
State
on

ok,00
```

set Arguments:

`<{on|off}>`	Turn the system IR receiver (all cameras) on or off.

set Examples:

```
set camera ir off

ok,00
```

lock

When used with the **get** verb, the **lock** target shows whether the PTZ camera lock mechanism is on (camera motors are locked in place) or off. When used with the **set** verb, this target enables or disables camera motor movement. You cannot move or zoom a PTZ camera that is locked. LifeSize Focus does not support camera control.

get Arguments:

> None

get Examples:

```
get camera lock
1,off

ok,00

get camera lock -V
Camera      Lock Mode
1           off

ok
```

set Arguments:

`[-N <{1|a}>]`	Optional: Specify the camera to lock. Only one camera is possible.
`<{on|off}>`	Enable or disable the lock.

set Examples:

set camera lock *on*

ok,00

set camera lock -N *1 off*

ok,00

lock-preset

When used with the **get** verb, the **lock-preset** target shows whether or not the PTZ camera presets are locked in memory. When used with the **set** verb, this target controls whether or not the presets are locked. When the presets are locked, they cannot be modified with the remote control or by the far end in a call. This does not affect setting presets with the **set camera preset** command. LifeSize Focus does not support camera control.

get Arguments:

None

get Examples:

get camera lock-preset
on

ok,00

get camera lock-preset -V
Preset Lock
off

ok

set Arguments:

<{on|off}>	Specify whether the preset lock is on or off.

set Examples:

set camera lock-preset *on*

ok,00

camera

pan-dir

When used with the **get** verb, the **pan-dir** target retrieves the current setting for the PTZ camera pan direction. This setting affects the direction the camera moves when using the left and right buttons on the remote control, but does not affect the **-l** and **-r** arguments of the **set camera position** command. When used with the **set** verb, this target determines whether the camera pans in the direction it is perceived by the user when the user is facing the camera, or in the reverse, from the camera's point of view. LifeSize Focus does not support camera control.

get Arguments:

> None

get Examples:

> **get camera pan-dir**
> *perceived*
>
> ok,00
>
> **get camera pan-dir -V**
> Mode
> *reversed*
>
> ok

set Arguments:

`<{perceived|reversed}>`	Specify the direction of pan.

set Examples:

> **set camera pan-dir** *perceived*
>
> ok,00

position

When used with the **get** verb, the **position** target retrieves the position of the specified PTZ camera. The default retrieves the position of the active camera. When used with the **set** verb, this target controls the current position of the near camera. Using the absolute position commands, you can recall a preset position. Using the motion commands, remote control of the camera is possible through press and release button mappings.
LifeSize Focus does not support camera control.

get Arguments:

`[-N <{1|a}>]`	Optional: Retrieves the position of the specified camera. Only one camera is possible.
`[-e]`	Retrieve the position in exact coordinates using floating point representation.

get Examples:

get camera position
-30,5,15

ok,00

get camera position -N *1*
45|5|35

ok,00

get camera position -V
Pan   Tilt   Zoom
45   -5   30

ok

get camera position -e
44.39,-5.02,29.87

ok,00

The ranges of the pan, tilt, and zoom values are shown in the following table. The camera may not be able to reach the maximum and minimum values for each parameter (for example, sending a command to the camera to pan to -45 may return a value other than -45). This is normal. Due to sensor resolution limits, one of the camera position parameters

may change slightly when another position is modified (for example, adjusting the tilt angle may slightly affect the pan position). The ranges allowed are designed for future compatibility; therefore, your camera may not reach the limits.

Camera	Pan	Tilt	Zoom
LifeSize PTZ	-45 to 45 degrees	-30 to 30 degrees	0 to 100

set Arguments:

`[-p <{-180.0..180.0}>]`	Specify the absolute pan angle. The range is -180 to 180 and represents the degrees left (negative) or right (positive) of the center position from which the camera pans. Use floating point notation to increase the precision of movement. Cannot be used with -r, -l, -u, -d, -n, -f, -s, or -c.
`[-t <{-90.0..90.0}>]`	Specify the absolute tilt angle. The range is -90 to 90 and represents the degrees below (negative) or above (positive) horizontal from which the camera tilts. Use floating point notation to increase the precision of movement. Cannot be used with -r, -l, -u, -d, -n, -f, -s, or -c.
`[-z <{0.0..100.0}>]`	Specify the absolute zoom position. The range is 0 to 100 and represents the range of the zoom lens from widest angle (0) to narrowest angle (100). Use floating point notation to increase the precision of movement. Cannot be used with -r, -l, -u, -d, -n, -f, -s, or -c.
`[-l]`	Specify that the camera pans to the left. Cannot be used with any other argument except -c.
`[-r]`	Specify that the camera pans to the right. Cannot be used with any other argument except -c.
`[-d]`	Specify that the camera tilts down. Cannot be used with any other argument except -c.
`[-u]`	Specify that the camera tilts up. Cannot be used with any other argument except -c.

[-n]	Specify the camera zooms in (telephoto). Cannot be used with any other argument except -c.
[-f]	Specify the camera zooms out (widens). Cannot be used with any other argument except -c.
[-s]	Specify the camera stop all movements. Cannot be used with any other argument except -c.
[-c n]	Specify that the camera control operation is applied to the far camera for the specified call. Cannot be used with -p, -t, or -z. The default controls the active near camera. Specifying the value 0 for the call controls the far end camera of the most recent dominant talker.
[-N <{1|a}>]	Optional: Specify a camera to apply the change. Only one camera is possible. Cannot be used with -c.
[-P <{0..19}>]	Specify the preset to use. Can only be used with the -N option when the preset 0 is selected. This command moves the camera indicated by the preset to the position specified by the preset.

If you specify the camera's current position or a position that is not significantly different from the camera's current position, the camera motors do not move. For example, if the camera is set to position 0.5 on pan or tilt and you change the value to 0.6, the increment is not large enough to invoke a change. The **set camera position** command in this case returns error, 02. To work around this issue, enter a value for the current setting that is incrementally larger than the value you wish to enter. Then re-enter the value you wish to enter to invoke the change.

Note: The camera motors may not move when you specify a position with the **set camera position** command due to other conditions. For example, the camera may be locked or initializing.

set Examples:

Note: The double dash (--) is not required in the following input because -15 is an argument to the option -t.

```
set camera position -p 34 -t -15

ok,00
```

camera

Start the camera panning left, then zoom in and stop all operations:

set camera position *-l*

ok,00

set camera position *-n*

ok,00

set camera position *-s*

ok,00

Start the far camera panning left in call 1:

set camera position –r –c *1*

ok,00

Note: The command completes immediately and does not wait for the camera to complete the operation.

Move to a specific preset position:

set camera position –P *12*

ok,00

Move to a specific position using floating point notation:

set camera position -p *15.32* -t *-14.3* -z *32.24*

ok,00

preset

When used with the **get** verb, the **preset** target retrieves the preset position information for the PTZ camera. The output may be restricted to a specific camera or a specific preset or set of presets. The preset position stores the associated camera and its pan, tilt, and zoom location. When used with the **set** verb, this target stores the preset positions for the camera in the codec's memory. Up to 19 different positions can be stored. Positions 1-9 are recallable through the remote control (in addition to the special position 0) and positions 10-19 are only recallable using the CLI. LifeSize Focus does not support camera control.

get Arguments:

`[-N <{1|a}>]`	Optional: Retrieves the position of the specified camera. Because only one camera is possible, camera `1` and camera `a` are effectively the same.
`[-P {0..19}]`	Retrieves a specific preset, which may be specified multiple times to retrieve several preset positions. Presets are listed in the same order as given on the command line.
`[-e]`	Retrieve the position using exact coordinates through floating point representation.

get Examples:

```
get camera preset
0,any,0,0,13
1,1,0,0,50
2,1,-20,10,35
3,1,10,-5,0

ok,00
```

```
get camera preset -V -P 3 -P 2 -P 0
Preset    Camera    Pan     Tilt     Zoom
3         1         10      -5       0
2         1         -20     10       35
0         any       0       0        13

ok
```

camera

```
get camera preset -N a
0,any,0,0,13
1,1,0,0,50
3,1,10,-5,0

ok,00

get camera preset -e
0,any,0.00,0.00,13.75
1,1,0.00,0.00,50.01
2,1,-19.57,10.23,0.45
3,1,9.87,-4.93,0.45

ok,00
```

set Arguments:

`[-N <{1|a)>]`	Optional: Specify the camera to which the preset applies. Because only one camera is possible, camera `1` and camera `a` (active camera) are effectively the same.
`-P <{1..19}>`	Specify the preset position to store.
`[-p <-180.0..180.0}>]`	Specify the pan position of the preset. The default is the current position of the selected camera. Use floating point notation to increase the precision of the movement.
`[-t <{-90.0..90.0}>]`	Specify the tilt position of the preset. The default is the current position of the selected camera. Use floating point notation to increase the precision of the movement.
`[-z <{0.0..100.0}>]`	Specify the zoom position of the preset. The default is the current position of the selected camera. Use floating point notation to increase the precision of the movement.

set Examples:

Set preset 2 to the camera's current position:

set camera preset -P *2*

ok,00

Set preset 4 to a specific position for camera 1:

set camera preset -P *4* **-N** *1* **-p** *28* **-t** *4* **-z** *20*

ok,00

Use floating point notation for more precision:

set camera preset -P *5* **-N** *1* **-p** *25.3* **-t** *14.9* **-z** *12.2*

ok,00

type

The **type** target shows the type of cameras connected to the system. This information appears on the **System Information** page in the user interface following status information about connected cameras as the value for the **Type** field. Possible values include:

- none
- ptz

  Pan, tilt, and zoom camera
- fixed_focus

  Fixed-focus camera with microphones

This target applies to the **get** verb.

Arguments:

None

Examples:

LifeSize Passport with LifeSize Focus:

get camera type
fixed_focus

ok,00

camera

```
get camera type -V
Camera 1
fixed_focus

ok
```

LifeSize Passport with LifeSize Camera:

```
get camera type
ptz

ok,00
```

white-balance

When used with the **get** verb, the **white-balance** target retrieves the current white balance setting for the cameras. When used with the **set** verb, this target sets the current white balance setting for the specified camera.

get Arguments:

> None

get Examples:

```
get camera white-balance
auto

ok,00
```

set Arguments:

[-N <{1|a}>]	Optional: Specify the camera to adjust. Because only one camera is possible, camera 1 and camera a (active camera) are effectively the same.
<{auto|indoor1| indoor2|outdoor| incan|incan-3200k| halogen}>	Set the white balance to the specified setting.

set Examples:

```
set camera white-balance halogen

ok,00
```

```
set camera white-balance indoor1
```

```
ok,00
```

conference

The following target applies to the **conference** object.

security

When used with the **get** verb, the **security** target reports whether or not encryption is required for conference calls. When used with the **set** verb, this target enables or disables the use of encryption for conference calls.

get Arguments:

    None

get Examples:

```
get conference security
off
```

```
ok,00
```

```
get conference security -V
H.235 Encryption
strict
```

```
ok
```

set Arguments:

`<{off｜on｜strict}>`	Specify *off* to disable encryption. Specify *on* to allow encryption. Specify *strict* to require encryption.

set Examples:

```
set conference security strict
```

```
ok,00
```

config

When used with the `get` verb, the `config` target retrieves the current configuration of the system. This includes all saved parameters currently configurable by the CLI. The output is in the form of a script suitable for execution by the CLI. Before using the script with the `set config` command to restore the configuration of a system, you must edit the script as follows:

- Stored passwords are replaced by tokens surrounded by '###' characters (e.g., ###`password`###). Replace the these characters and tokens with the password.

- The system must be rebooted after the configuration is applied. Delete the trailing `ok,00` from the end of the script if it was captured. Append `control reboot` to the end of the script to effect a reboot.

When used with the `set` verb, this target allows reloading the system configuration from a script produced by `get config`. This is an alternate method to reading that script. It can also be fed directly to the CLI. The advantages are that the output of this command indicates the line numbers of failing commands in the script and the error codes of those commands and will exit with a return code indicating whether the entire script failed or succeeded. With the direct input method, the error messages for failing commands are mixed in with the output, and the exit code is that of the last command executed.

get Arguments:

`[-P]`	Export the file with all passwords except the shell password and SNMP passwords. If you use this argument, all other passwords are visible without the pound (#) symbols.

get Examples:

```
sh% ssh auto@ip get config > codec_confg.as
sh%
```

To restore the configuration to a system using the direct input method:

```
sh% ssh auto@otherip < codec_config.as
command 1

ok,00
....
```

The output of the restore using the direct input method lists the executed commands followed by the return status of the command. Any command failures are indicated in the normal way. The script execution does not stop due to intermediate failures and the exit status of the script is the status of the final command that is executed.

set Arguments:

[-i]	Ignore errors in the script and execute to the end. The default executes up to the first error and then stops.

set Examples:

```
unix% ssh auto@ip get config > script
unix% vi script # fix up passwords
unix% ssh auto@otherip set config -i -V < script
```

```
Line      Error      Note      Command

23        09         FIX       set admin password ###password###

325       09                   set video secondary-background "Image"


error,09
```

directory

The following targets apply to the **directory** object.

auto

When used with the **get** verb, the **auto** target retrieves the configuration for the auto discovery daemon. When used with the **set** verb, this target controls configuration of the auto discovery daemon.

get Arguments:

> None

get Examples:

```
get directory auto
On,10.10.11.* 10.10.10.*,192.168.*


ok,00
```

```
get directory auto -V
State           Searched Subnets         Ignored Subnets
On              10.10.11.* 10.10.10.*    192.168.*


ok
```

directory

set Arguments:

`[-a]`	Append `-i` and `-s` options to the appropriate lists instead of replacing the lists.
`[-i ipaddr]`	Specify a subnet to ignore during discovery. For example, 10.10.11.* ignores all devices with an IP address of 10.10.11.0 through 10.10.11.255.
`[-s ipaddr]`	Specify a subnet to search during discovery. For example, 10.10.* searches all devices with an IP address of 10.10.0.0 through 10.10.255.255.
`<{on|off}>`	Enable or disable the auto discovery daemon.

Note: By default, the auto discovery daemon searches only the subnet defined by its IP address and network mask.

set Examples:

Discover on 10.10.11 and 10.10.12, ignore 10.10.10:

set directory auto -s *10.10.11.* -s *10.10.12.* -i *10.10.10.* on*

ok,00

Add discovery on 10.10.13:

set directory auto -s *10.10.13.* -a on*

ok,00

Reset discovery to 10.10.13 only, clearing ignored subnets too:

set directory auto -s *10.10.13.* on*

ok,00

Turn off discovery entirely:

set directory auto *off*

ok,00

corporate

The **corporate** target retrieves the corporate directory entries. These entries come from either the LDAP server's directory if LDAP is enabled, or through auto discovery if auto discovery is enabled. This target applies to the **get** verb.

Arguments:

`[-1 number]`	Restrict the listing to those entries whose Number column starts with the specified ISDN number.
`[-B]`	Send the data in a format suitable for processing by the **set directory local -B** command.
`[-C]`	Show only the number of entries matching the selection criteria.
`[-H hier]`	Restrict the listing to those entries whose Hierarchy column starts with the specified hierarchy grouping.
`[-a isdn-ac]`	Restrict the listing to those entries whose AC column starts with the specified ISDN area code.
`[-c isdn-cc]`	Restrict the listing to those entries whose CC column starts with the specified ISDN country code.
`[-i ip-addr]`	Restrict the listing to those entries whose IP Address column starts with the specified IP address.
`[-n count]`	Limit the output to `count` entries.
`[-s count]`	Skip the first count entries.
`[-t phone]`	Restrict the listing to those entries whose Voice column starts with the specified voice telephone number.
`[-v video]`	Restrict the listing to those entries whose Video column starts with the specified video number.
`[prefix]`	Restrict the listing to those entries starting with the specified prefix (case insensitive).

directory

Examples:

get directory corporate
John Doe,,10.10.11.254,,,,,,,auto,auto
Mary Jane,,10.10.11.213,,,,,,,auto,auto
Noah James,,10.10.11.116,,,,,,,auto,auto
Steve Jones,,10.10.11.155,,,,,,,auto,auto
Test,,10.10.11.8,,,,,,,auto,auto

ok,00

get directory corporate –V

Name	Voice	Video	IP Address	CC	AC	Number	X	Hierarchy	IP BW	ISDN BW
John Doe		*10.10.11.254*							*auto*	*auto*
Mary Jane		*10.10.11.213*							*auto*	*auto*
Noah James		*10.10.11.116*							*auto*	*auto*
Steve Jones		*10.10.11.155*							*auto*	*auto*
Test		*10.10.11.8*							*auto*	*auto*

ok

Note: Field X is reserved and contains no data.

get directory corporate *n*
Noah James,,10.10.11.116,,,,,,,auto,auto

ok,00

get directory corporate -B
John Doe||10.10.11.254||||||auto|auto
Mary Jane||10.10.11.213||||||auto|auto
Noah James||10.10.11.116||||||auto|auto
Steve Jones||10.10.11.155||||||auto|auto
Test||10.10.11.8||||||auto|auto

ok,00

get dir corporate -C -V
Count
5

ok

Specification of multiple selection options results in output that is the logical and for all the conditions (each line must match all conditions specified). Regular expressions in and logical ordering of the selection criteria are not supported. The **-n** and **-s** arguments allow remote programs to page through the directory entries by limiting the range of entries shown.

ldap

When used with the **get** verb, the **ldap** target shows the configuration of the LDAP directory settings for the corporate directory. When used with the **set** verb, this target configures the LDAP directory server preferences.

get Arguments:

   None

get Examples:

   get directory ldap
   Off,,,,,,never,unregistered

   ok,00

   get directory ldap -V
   State   Server   Username   Base   Filter   Refresh   Status
   Off   unregistered

   ok

set Arguments:

-d	Disable LDAP.
-e	Enable LDAP.
[-f *filter*]	Specify the LDAP server search filter.
[-o *ou-value*]	Specify the value of the OU field. Default is *EndPoints*.
[-p *password*]	Specify the password used to access the LDAP server.

directory

`[-r {1-minute|5-minutes|`
`10-minutes|20-minutes|`
`30-minutes|1-hour|`
`2-hours|3-hours|`
`6-hours|12-hours|`
`1-day|1-week|never}]`
	Specify the time interval between LDAP server refreshes.

`[-s host-or-ip]`	Specify the hostname or IP address of the LDAP server.
`[-u username]`	Specify the username used to access the LDAP server.
`[dc-value]`	Specify the values of the DC fields in the order of use. The OU field is hard coded as `EndPoints`.

set Examples:

**set directory ldap -e -u *user* -p *password* -s *ldap-server LifeSize*
   *VideoCodec***

ok,00

set directory ldap -r *1-hour*

ok,00

local

When used with the **get** verb, the **local** target retrieves the local directory entries. When used with the **set** verb, this target edits the local directory and supports the use of here documents to load the directory. For more information about here documents, refer to "Here Documents" on page 11.

get Arguments:

`[-1 number]`	Restrict the listing to those entries whose Number column starts with the specified ISDN number.
`[-B]`	Send the data in a format suitable for processing by the **set directory local -B** command.
`[-C]`	Show only the number of entries matching the selection criteria.

`[-H hier]`	Restrict the listing to those entries whose Hierarchy column starts with the specified hierarchy grouping.
`[-a isdn-ac]`	Restrict the listing to those entries whose AC column starts with the specified ISDN area code.
`[-c isdn-cc]`	Restrict the listing to those entries whose CC column starts with the specified ISDN country code.
`[-i ip-addr]`	Restrict the listing to those entries whose IP Address column starts with the specified IP address.
`[-n count]`	Limit the output to count entries.
`[-s count]`	Skip the first count entries.
`[-t phone]`	Restrict the listing to those entries whose Voice column starts with the specified voice telephone number.
`[-v video]`	Restrict the listing to those entries whose Video column starts with the specified video number.
`[prefix]`	Restrict the listing to those entries starting with the specified prefix (case insensitive).

get Examples:

```
get directory local
John Doe,,,10.10.11.254,,,,Video,auto,auto
Mary Jane,,,10.10.11.213,,,,Video,auto,auto
Noah James,,,10.10.11.116,,,,Video,1024,auto
Steve Jones,,,10.10.11.155,,,,Video,auto,auto
Test,555.1212,10.10.11.8,1.2.3.4,52,215,5550199,,Audio,auto,128

ok,00
```

directory

```
get directory local –V
```

Name	Voice	Video	IP Address	CC	AC	Number	X	Hierarchy	IP BW	ISDN BW
John Doe			*10.10.11.254*					*Video*	*auto*	*auto*
Mary Jane			*10.10.11.213*					*Video*	*auto*	*auto*
Noah James			*10.10.11.116*					*Video*	*1024*	*auto*
Steve Jones			*10.10.11.155*					*Video*	*auto*	*auto*
Test	*555.1212*	*10.10.11.8*	*1.2.3.4*	*52*	*215*	*5550199*		*Audio*	*auto*	*128*

ok

Note: Field X is reserved and contains no data.

```
get directory local n
```
Noah James,,,10.10.11.116,,,,Video,1024,auto

ok,00

```
get directory local -B
```
John Doe|||10.10.11.254|||Video|auto|auto
Mary Jane|||10.10.11.213|||Video|auto|auto
Noah James|||10.10.11.116|||Video|1024|auto
Steve Jones|||10.10.11.155|||Video|auto|auto
Test|555.1212|10.10.11.8|1.2.3.4|52|215|5550199||Audio|auto|128

ok,00

```
get dir local -C -V
```
Count
5

ok

Specification of multiple selection options results in output that is the logical and for all the conditions (each line must match all conditions specified). Regular expressions in and logical ordering of the selection criteria are not supported. The **-n** and **-s** arguments allow remote programs to page through the directory entries by limiting the range of entries shown.

set Arguments:

`[-1 number]`	Set the ISDN number.
`[-2 x]`	This argument is deprecated. The column labeled `B2 Number` in the output in previous releases appears with the column label `X` and is unused.
`[-B]`	Read data from standard input in batch mode. Accepts either the output of get local directory `-B` or the result of exporting the directory through the web administration interface. This argument cannot be used with any of the field set options.
`[-H hier]`	Specify the hierarchy for the entry. Use commas to delimit the hierarchy levels.
`[-K {auto|128|192|256| 320|384|512|640|768| 896|1024|1152|1280| 1472|1728|1920}]`	Set the bandwidth limit in kilobits per second (kb/s) for ISDN calls. The default is `auto`.
`[-a isdn-ac]`	Set the ISDN area code.
`[-c isdn-cc]`	Set the ISDN country code.
`[-d]`	Delete the specified entries. Only the name argument is used in single or batch mode. Matching entries are removed from the directory. Entries must match exactly (except for case).
`[-i ip-addr]`	Set the IP address for calls (used as backup if the video or telephone numbers are not specified.)
`[-k {auto|128|192|256| 320|384|512|640|768| 896|1024|1152|1280| 1472|1728|1920|2000}]`	Set the bandwidth limit in kilobits per second for video calls. The default is `auto`.
`[-o]`	Overwrite existing entries. The default leaves existing entries unchanged.
`[-t phone]`	Set the voice telephone number.

directory

`[-u]`	Update existing entries by merging new data with old data. The default leaves existing entries unchanged.
`[-v video]`	Set the video telephone number.
`name`	Set the name of the entry. Case is preserved, but is not significant in locating a matching entry. Cannot be used in batch mode.

set Examples:

Copy the local directory from one system to another:

```
sh$ ssh auto@room1 get directory local -B > localdir.txt
sh$ ssh auto@room2 set directory local -o -B < localdir.txt
```

Add an entry for "Sunbob" to the local directory stored hierarchically under Sun and then bob:

```
set directory local -i 10.10.10.11 -H Sun,bob Sunbob
```

```
ok,00
```

Update the Sunbob entry to include a voice number:

```
set directory local -t 5551212 -u Sunbob
```

```
ok,00
```

Overwrite the sunbob entry removing the voice number:

```
set directory local -i 10.10.10.11 -H Sun,bob -o Sunbob
```

```
ok,00
```

Alternative:

```
set directory local -t "" -u Sunbob
```

```
ok,00
```

Delete the Sunbob entry:

```
set directory local -d Sunbob
```

```
ok,00
```

meeting

When used with the **get** verb, the **meeting** target retrieves directory entries for the meetings directory. Because meetings contain multiple participants per entry, the output of the command is modal. By default, the output consists of the selected meetings with participant counts and meeting type. The -1 argument lists the individual participants of a single specified meeting. When used with the **set** verb, this target enables you to create or edit meeting directory entries and supports the use of here documents to load the directory. For more information about here documents, refer to "Here Documents" on page 11.

get Arguments:

[-B]	Output directory in batch mode suitable for importing with the **set directory meeting -B** command.
[-H hierarchy]	Select meetings from within the specified hierarchy only. Use a comma to separate levels of the hierarchy.
[-l]	List meeting participants. Cannot be used with **-n** or **-s**. You must specify only one meeting with this argument.
[-n count]	Limit meeting display to at most count entries. Cannot be used with **-l**.
[-s count]	Skip the first count entries. Useful for paging when used with **-n**. Cannot be used with **-l**.
[meeting]	Select meetings based on the meeting string (case indifferent, string must match start of meeting name). When used with **-l**, specify only one meeting.

get Examples:

```
get directory meeting
BoD Meeting,2,John Smith  Big Room,B,Video
Sales Meeting,4,Jim  Bob  Jan  Joe,S,Audio

ok,00

get directory meeting -V
Meeting Name    Number   Participants         Hierarchy    Type
BoD Meeting     2        John Smith, Big Room B            Video
Sales Meeting   4        Jim, Bob, Jan, Joe   S            Audio

ok
```

directory

get directory meeting -l *BOD*
John Smith,10.10.10.11,,,auto,Video
Big Room,10.10.10.12,,,auto,Video

ok,00

get directory meeting -V -l *BO*

Name	Video	Audio	IP Address	Bandwidth	Type
John Smith	*10.10.10.11*			*auto*	*Video*
Big Room	*10.10.10.12*			*auto*	*Video*

ok

get directory meeting -l

error,10,Ambiguous Selection

get directory meeting -H *b*
BoD Meeting,2,John Smith Big Room,B,Video

ok,00

get directory meeting -B
BoD Meeting|2|John Smith, Big Room|B|Video|Meeting
John Smith|10.10.10.11|||auto|Video
Big Room|10.10.10.12|||auto|Video
Sales Meeting|4|Jim, Bob, Jan, Joe|S|Audio|Meeting
Big Room|10.10.10.12|||auto|Audio
Tokyo|209.154.11.13|||auto|Audio
London|145.223.231.33|||auto|Audio
Boston|111.222.33.44|||auto|Audio

ok,00

set Arguments:

`[-B]`	Read standard input and create meetings based on batch mode data from **`get directory meeting -B`**.
`[-o]`	Overwrite existing entries. The default leaves existing entries unchanged. This argument cannot be used with any option other than `-B`.
`[-H hierarchy]`	Specify the hierarchy for the meeting. Use a comma to separate levels.
`[-r party]`	Specify a party to remove from a meeting. This argument is only valid when editing an existing meeting or creating a meeting copy.
`[-d]`	Delete the named meeting. In batch mode, deletes the meetings listed in the input file.
`[-m name]`	Specify the name of the meeting to create or modify. If the meeting already exists, parties are added to it. If not, it is created. If used with the `-d` argument, the meeting must already exist or the command fails. This option is required unless using batch mode.
`[-c name]`	Copy the specified meeting into this new meeting entry. The copy meeting will replace any existing meeting by that name.
`[-a party]`	Add the named party as a voice call. The party name can be any valid dial string. For examples, refer to **`control call dial`** in "dial" on page 223.
`[-i party]`	Add the named party as a video call using the ISDN number specified in the directory entry.
`[-v party]`	Add the named party as a video call. The party name can be any valid dial string.
`[party]`	Add the named party as a video or voice call based on the type specified in the directory or redial entry. If you do not specify the call type, the system automatically attempts to determine the type using the following order: video, voice, and ISDN.

directory

set Examples:

Create a new meeting ("QBR Meeting" does not exist):

set directory meeting -H *Q,M* **-m** *"QBR Meeting"* **-v** *"local:Big Room"*
   -v *local:Bangalore*

ok,00

Add a PSTN conference bridge to the meeting:

set directory meeting -m *"QBR Meeting"* **-a** *555-1212*

ok,00

Make a copy of the "QBR Meeting" adding one more party:

set directory meeting -c *"QBR Meeting"* **-H Y,M -m** *"YBR Meeting"*
   corp:Investors

ok,00

Make a copy of the "YBR Meeting" removing "Big Room" and adding Cube:

set directory meeting -c *"YBR Meeting"* **-m** *"India"* **-H I -r** *"Big*
   Room" **-i** *Cube*

ok,00

Remove the Bangalore conference room from the meeting:

set directory meeting -m *"QBR Meeting"* **-r** *Bangalore*

ok,00

Delete the "QBR Meeting":

set directory meeting -d -m *"QBR Meeting"*

ok,00

h323

The following targets apply to the **h323** object.

alternate

When used with the **get** verb, the **alternate** target retrieves the current settings for the alternate H.323 gatekeeper. When used with the **set** verb, this target configures the settings for the alternate H.323 gatekeeper when in manual mode.

get Arguments:

> None

get Examples:

> **get h323 alternate**
> *10.10.11.12,1719*
>
> ok,00
>
> **get h323 alternate -V**
> IP Address   Port
> *10.10.11.110   12345*
>
> ok

set Arguments:

ipaddr	Specify the IP address for the gatekeeper in manual mode.
[*port*]	Optional: Specify the port for the gatekeeper. The default is 1719 or the current setting.

set Examples:

> **set h323 alternate** *10.10.11.12*
>
> ok,00
>
> **set h323 alternate** *10.10.11.12 1832*
>
> ok,00

extension

When used with the **get** verb, the **extension** target retrieves the H.323 extension associated with the endpoint. When used with the **set** verb, this target sets the extension to use when registering the device with the H.323 gatekeeper.

get Arguments:

> None

get Examples:

```
get h323 extension
1188

ok,00

get h323 extension -V
Extension
1188

ok
```

set Arguments:

extension	Specify the extension to use when registering with the H.323 gatekeeper.

set Examples:

```
set h323 extension 1188

ok,00
```

gkauth

When used with the **get** verb, the **gkauth** target reveals whether gatekeeper authentication is enabled or disabled. When enabled, you must provide a gatekeeper username and password to use that gatekeeper. When used with the **set** verb, this target enables or disables gatekeeper authentication.

get Arguments:

    None

get Examples:

    get h323 gkauth
    disabled

    ok,00

    get h323 gkauth -V
    Authentication
    enabled

    ok

set Arguments:

`<{enabled|disabled}>`	Specify whether to enable or disable gatekeeper authentication.

set Examples:

    set h323 gkauth *enabled*

    ok,00

gkid

When used with the **get** verb, the **gkid** target retrieves the gatekeeper username used in gatekeeper authentication. When used with the **set** verb, this target provides gatekeeper authorization username.

get Arguments:

None

get Examples:

```
get h323 gkid
username

ok,00

get h323 gkid -V
Username
username

ok
```

set Arguments:

username	Specify the gatekeeper authentication username.

set Examples:

```
set h323 gkid username

ok,00
```

gkpass

Use this target to enter the gatekeeper authorization password. This target applies to the **set** verb.

set Arguments:

password	Specify the gatekeeper authentication password.

set Examples:

```
set h323 gkid password

ok,00
```

h323

When used with the **get** verb, the **h323** target shows whether H.323 calls are enabled or disabled. When used with the **set** verb, this target controls whether H.323 calls are enabled or disabled.

get Arguments:

>   None

get Examples:

```
get h323 h323
enabled

ok,00


get h323 h323 -V
H323
enabled

ok
```

set Arguments:

<{enabled|disabled}>	Specify whether to enable or disable H.323 calls.

set Examples:

```
set h323 h323 disabled

ok,00
```

h460

When used with the **get** verb, the **h460** target shows whether support for H.460 with H.323 calls is enabled or disabled. When used with the **set** verb, this target controls whether support for H.460 with H.323 calls is enabled or disabled.

get Arguments:

>   None

get Examples:

```
get h323 h460
enabled

ok,00
```

h323

```
get h323 h460 -V
H460
enabled

ok
```

set Arguments:

`<{enabled|disabled}>`	Specify whether to enable or disable H.460 support with H.323 calls.

set Examples:

```
set h323 h460 disabled

ok,00
```

id

When used with the **get** verb, the **id** target retrieves the H.323 gatekeeper ID. When used with the **set** verb, this target sets the H.323 gatekeeper ID.

get Arguments:

None

get Examples:

```
get h323 id
RADGK

ok,00
```

```
get h323 id -V
Gatekeeper ID
RADGK

ok
```

set Arguments:

`id`	Specify the gatekeeper ID.

set Examples:

```
set h323 id RADGK

ok,00
```

mode

When used with the **get** verb, the **mode** target retrieves the H.323 gatekeeper mode, which indicates whether the gatekeeper is used at all or manually or automatically configured. When used with the **set** verb, this target configures the H.323 gatekeeper mode.

get Arguments:

   None

get Examples:

```
get h323 mode
off

ok,00

get h323 mode -V
Mode
manual

ok
```

set Arguments:

`<{off|manual|auto}>`	Specify the gatekeeper mode. The `off` argument disables use of the H.323 gatekeeper, `manual` uses the primary and alternate settings, and `auto` determines the gatekeeper information automatically.

set Examples:

```
set h323 mode auto

ok,00

set h323 mode off

ok,00
```

name

When used with the **get** verb, the **name** target retrieves the currently configured H.323 name for the device. When used with the **set** verb, this target sets the H.323 name for the device.

get Arguments:

None

get Examples:

get h323 name
Conference Room950

ok,00

get h323 name -V
Name
Conference Room950

ok

set Arguments:

name	Specify the name to use for the device when registering with the H.323 gatekeeper.

set Examples:

set h323 name *LifeSize*

ok,00

primary

When used with the **get** verb, the **primary** target retrieves the configuration for the H.323 primary gatekeeper. When used with the **set** verb, this target configures the H.323 primary gatekeeper when the primary gatekeeper is in manual mode.

get Arguments:

None

get Examples:

get h323 primary
10.10.11.12,1719

ok,00

```
get h323 primary -V
```
```
IP Address       Port
```
10.10.11.110   12345

```
ok
```

set Arguments:

ipaddr	Specify the IP address for the gatekeeper in manual mode.
[*port*]	Optional: Specify the port for the gatekeeper. The default is 1719 or the current setting.

set Examples:

```
set h323 primary 10.10.11.12 1719
```

```
ok,00
```

```
set h323 primary 10.10.11.15
```

```
ok,00
```

register

When used with the `get` verb, the `register` target retrieves the current registration status of the H.323 gatekeeper. When used with the `set` verb, this target starts the registration process with the configured H.323 gatekeeper. Because registration may take an arbitrarily long time, the command returns immediately. Use the `get h323 register` command to check the status.

get Arguments:

None

get Examples:

```
get h323 register
registered
```

```
ok,00
```

get h323 register -V
Status
failed

ok

set Arguments:

None

set Examples:

set h323 register

ok,00

help-mode

When used with the **get** verb, the **help-mode** target retrieves the current setting for help mode. When used with the **set** verb, this target controls whether or not help is available. It also enables and disables the use of abbreviations for commands (abbreviating **help-mode** as just **help**). To avoid ambiguity in future software releases, LifeSize recommends that you do not use abbreviations in scripts.

get Arguments:

None

get Examples:

get help-mode
on

ok,00

set Arguments:

<{on|off}>	Enable or disable help and abbreviation mode.

set Examples:

set help-mode *on*

ok,00

http

When used with the **get** verb, the **http** target shows whether the web (HTTP) service is enabled or disabled. When used with the **set** verb, this target controls whether the web (HTTP) service is enabled or disabled.

get Arguments:

    None

get Examples:

    get http
    on

    ok,00

    get http -V
    Web (http) Service
    off

    ok

set Arguments:

<{off|on}>	Disable or enable the http service.

set Examples:

    set http *on*

    ok,00

locale

The **locale** object controls location-specific information for a device. The following targets apply to the **locale** object.

country

When used with the **get** verb, the **country** target shows the current country setting for the system. When used with the **set** verb, this target configures the country setting.

get Arguments:

> None

get Examples:

> **get locale country**
> *algeria*
>
> ok,00
>
> **get locale country -V**
> Country
> *uruguay*
>
> ok

set Arguments:

`<{algeria|argentina|australia|austria|bahrain|`
`belarus|belgium|brazil|brunei|bulgaria|canada|`
`chile|china|columbia|croatia|cyprus|`
`czech-republic|denmark|ecuador|egypt|estonia|`
`finland|france|germany|ghana|greece|`
`hong-kong|hungary|india|indonesia|ireland|israel|`
`italy|cote-d-ivoire|japan|jordan|kazakhstan|`
`latvia|lebanon|lesotho|lithuania|luxembourg|`
`malaysia|malta|mexico|morocco|netherlands|`
`new-zealand|norway|oman|pakistan|paraguay|peru|`
`philippines|poland|portugal|puerto-rico|qatar|`
`romania|russia|singapore|slovakia|slovenia|`
`south-africa|south-korea|spain|sri-lanka|`
`sweden|switzerland|taiwan|thailand|tunisia|turkey|`
`ukraine|united-arab-emirates|united-kingdom|`
`united-states|uruguay|venezuela|vietnam|zambia}>`
	Specify the country code to use.

set Examples:

> **set locale country *algeria***
>
> ok,00

gmt-offset

The **gmt-offset** target retrieves the current Greenwich Mean Time (GMT) offset value for the time zone currently selected for the system. The format of the output for this command appears as *+HHMM* or *-HHMM* where *HH* is hours and *MM* is minutes. This target applies to the **get** verb.

get Arguments:

> None

get Examples:

> **get locale gmt-offset**
> *-0500*
>
> ok,00
>
> **get locale gmt-offset -V**
> GMT Offset
> *-0500*
>
> ok

language

When used with the **get** verb, the **language** target shows the current language used for user interface prompts and messages. When used with the **set** verb, this target sets the language used for user interface prompts and messages. This setting does not affect the input or output of the CLI.

get Arguments:

> None

get Examples:

> **get locale language**
> *german*
>
> ok,00

locale

```
get locale language -V
GUI Language
traditional-chinese

ok
```

set Arguments:

<{german|us-english|spanish|french|italian|japanese| korean|norwegian|brazilian-portuguese|russian| suomi-finnish|swedish|simplified-chinese| traditional-chinese|polish}>	Select the user interface language.

set Examples:

```
set locale language brazilian-portuguese

ok,00
```

timezone

When used with the **get** verb, the **timezone** target shows the current system time zone. When used with the **set** verb, this target changes the current system time zone.

get Arguments:

None

get Examples:

```
get locale timezone
majuro

ok,00
```

```
get locale timezone -V
Time Zone
kirimati

ok
```

set Arguments:

`<{majuro|midway|honolulu|anchorage|`
`los-angeles|vancouver|denver|edmonton|phoenix|`
`austin|guatemala-city|managua|mexico-city|`
`san-salvador|tegucigalpa|winnipeg|bogota|havana|`
`indianapolis|kingston|lima|montreal|nassau|`
`new-york|asuncion|caracas|halifax|la-paz|santiago|`
`santo-domingo|san-juan|st-johns|sao-paulo|`
`buenos-aires|montevideo|mid-atlantic|ponta-delgada|`
`greenwich-mean-time|dublin|lisbon|london|reykjavik|`
`amsterdam|berlin|brussels|budapest|copenhagen|`
`madrid|oslo|paris|prague|rome|stockholm|vienna|`
`warsaw|zagreb|zurich|athens|beirut|cairo|helsinki|`
`istanbul|jerusalem|johannesburg|kyiv|baghdad|`
`kuwait-city|moscow|riyadh|tehran|abu-dhabi|kabul|`
`almaty|karachi|new-delhi|kathmandu|dhaka|yangon|`
`bangkok|jakarta|beijing|kuala-lumpur|manila|perth|`
`singapore-city|taipei|seoul|tokyo|darwin|adelaide|`
`brisbane|sydney|vladivostok|suva|kamchatka|`
`wellington|chatham-island|kiritimati}>`
	Select the local timezone.

set Examples:

`set locale timezone greenwich-mean-time`

`ok,00`

network

The **network** object controls the current network configuration. If you use the **set** verb with a command that contains the **network** object, you must issue the **set network commit** command to commit the change. The **commit** target commits the network settings. Network settings that are changed but not committed do not take effect until the next system reboot. Some network commands when followed by the set **network commit** command cause the system to reboot. For a list of these commands, refer to "commit" on page 106.

Note: The **get network** command has been deprecated and superseded by the **get network ipv4** and **get network ipv6** commands.

The following targets apply to the **network** object.

802.1X

LifeSize video communications systems support port-based mutual authentication based on the IEEE 802.1X standard using the EAP-TLS sub-protocol. The **802.1x** object controls the use of this feature. The IEEE 802.1X standard provides port-based authentication involving communications between a supplicant, an authenticator (an 802.1X-capable Ethernet switch in this application), and an authentication server. The LifeSize codec attached to an 802.1X-controlled port on the switch performs the supplicant role. A back-end authentication server (typically, a RADIUS server) attached to a non-802.1X port on the switch usually performs the authentication server role. EAP packets flow between the supplicant (the codec) and the authenticator (the switch), and RADIUS packets flow between the authenticator (switch) and the authentication server (RADIUS server). Initially, 802.1X ports allow only 802.1X traffic; all other packets are blocked at the data link layer until the device attached to the port is authenticated.

This implementation assumes that you have configured the authentication server. The authentication server must have the CA certificate, the server certificate, and the server certificate private key installed. The server software must be configured with the location of the certificate and private key files, and with the text of the server certificate private key passphrase.

The authenticator must be configured to allow one or more of its ports to provide 802.1X access control, and it must be configured to access the authentication server.

This implementation also assumes that a certificate authority has produced a CA certificate, a client certificate, a client key and a client key passphrase for the LifeSize system. Before you enable this feature by setting the **state** target to *enabled*, you must first set the **ca-cert**, **client-cert**, **client-key**, and **client-key-passphrase** targets.

ca-cert

When used with the **get** verb and the **802.1x** object, the **ca-cert** target returns either the CA certificate set for the LifeSize system or an error if no CA certificate has been set. When used with the **set** verb and the **802.1x** object, this target adds a CA certificate to the system.

get Arguments:

> None

get Examples:

> **get network 802.1x ca-cert**
> *<certificate data>*
>
> ok,00
>
> **get network 802.1x ca-cert -V**
> CA Certificate
> *<certificate data>*
>
> ok

set Arguments:

> None

set Examples:

Certificate data can be manually entered in a here document.

> **set network 802.1x ca-cert << EOF**
> *certificate data*
> **EOF**
>
> ok,00

Certificate file data can be piped to the command if executed from a remote host.

> **ssh auto@*<ip address>* set network 802.1x ca-cert < *ca_cert.pem***

client-cert

When used with the **get** verb and the **802.1x** object, the **client-cert** target returns either the client certificate set for the LifeSize system or an error if no client certificate has been set. When used with the **set** verb and the **802.1x** object, this target adds a client certificate to the system.

get Arguments:

 None

get Examples:

```
get network 802.1x client-cert
<certificate data>

ok,00

get network 802.1x client-cert -V
Client Certificate
<certificate data>

ok
```

set Arguments:

 None

set Examples:

Certificate data can be manually entered through a here document as in the following example.

```
set network 802.1x client-cert << EOF
certificate data
EOF

ok,00
```

Certificate file data can be redirected to the command if executed from a remote host, as in the following example.

```
ssh auto@<ip address> set network 802.1x client-cert <
    client_cert.pem
```

client-key

When used with the **get** verb and the **802.1x** object, the **client-key** target returns either the client key set for the LifeSize system or an error if no client key has been set. When used with the **set** verb and the **802.1x** object, this target adds a client key to the system.

get Arguments:

> None

get Examples:

```
get network 802.1x client-key
<key data>

ok,00

get network 802.1x client-key -V
Client Private Key
<key data>

ok
```

set Arguments:

> None

set Examples:

The client key can be manually entered through a here document as in the following example.

```
set network 802.1x client-key << EOF
<client key>
EOF

ok,00
```

Certificate file data can be redirected to the command if executed from a remote host, as in the following example.

```
ssh auto@<ip address> set network 802.1x client-key <
    client_key.pem
```

client-key-passphrase

This target is write only. When used with the `set` verb and the `802.1x` object, the `client-key-passphrase` target adds a client key passphrase to the system.

set Arguments:

　　None

set Examples:

　　set network 802.1x client-key-passphrase *abcdef19!*

　　ok,00

state

When used with the `get` verb and the `802.1x` object, the `state` target shows whether support for 802.1X authentication is *enabled* or *disabled* on the video communications system codec. When used with the `set` verb and the `802.1x` object, this target controls whether support for 802.1X authentication is *enabled* or *disabled* on the video communications system codec. The default is ***disabled***.

Note: Before you set `state` to ***enabled***, you must set the `ca-cert`, `client-cert`, `client-key` and `client-key-passphrase` targets. You can also enable and disable 802.1X support through the system's user or web administration interfaces, but only after setting the `ca-cert`, `client-cert`, `client-key` and `client-key-passphrase` targets.

get Arguments:

　　None

get Examples:

　　get network 802.1x state
　　disabled

　　ok,00

　　get network 802.1x state -V
　　State
　　disabled

　　ok

set Arguments:

`<{enabled|disabled}>`	Enable or disable the feature.

set Examples:

```
set network 802.1x state disabled
```

```
ok,00
```

Commit the change:

```
set network commit
```

```
ok,00
```

amx-enable

When used with the **get** verb, the **amx-enable** target shows whether support for AMX Device Discovery through IP connectivity is enabled (*on*) or disabled (*off*) on the codec. When used with the **set** verb, this target controls whether support for AMX Device Discovery through IP connectivity is enabled or disabled on the codec. The default is *off*.

When support for AMX Device Discovery through IP connectivity is enabled, the codec transmits an IP beacon message as a UDP packet to the IP address 239.255.250.250 on port 9131 every 60 seconds. Information sent in the message includes the serial number, make, and model of the codec, the revision number of the AMX interface that supports the codec, and the class of the device as defined by AMX.

get Arguments:

None

get Examples:

```
get network amx-enable
off
```

```
ok,00
```

```
get network amx-enable -V
AMX Enabled
off
```

```
ok
```

set Arguments:

`<{on|off}>`	Enable or disable the feature.

set Examples:

`set network amx-enable` *`off`*

`ok,00`

Commit the change:

`set network commit`

`ok,00`

amx-make

The **`amx-make`** target retrieves the make of the device. The device includes this information in the IP beacon message that it transmits when **`amx-enable`** is set to *on*. This target applies to the **`get`** verb.

get Arguments:

None

get Examples:

`get network amx-make`
`LIFESIZE`

`ok,00`

`get network amx-make -V`
`AMX Beacon Make`
`LIFESIZE`

`ok`

Commit the change:

`set network commit`

`ok,00`

amx-master

When used with the **get** verb, the **amx-master** target retrieves the IP address to which the codec sends the IP beacon message when the **amx-enable** target is set to *on*. The default is 239.255.250.250. When used with the **set** verb, this target sets the IP address to which the codec sends the IP beacon message when the **amx-enable** target is set to *on*.

get Arguments:

　　None

get Examples:

```
get network amx-master
10.0.1.254

ok,00

get network amx-master -V
AMX Master IP Address
10.0.1.254

ok
```

set Arguments:

<IP address>	Specify the IP address to which the codec sends the IP beacon message when the **amx-enable** target is set to *on*.

set Examples:

```
set network amx-master 10.0.1.254

ok,00
```

Commit the change:

```
set network commit

ok,00
```

amx-model

The **amx-model** target retrieves the device model that the device sends in the IP beacon when the **amx-enable** target is set to *on*. This target applies to the **get** verb.

get Arguments:

　　None

network

get Examples:

```
get network amx-model
PASSPORT

ok,00

get network amx-model -V
AMX Beacon Model
PASSPORT

ok
```

amx-port

When used with get `verb`, the `amx-port` target retrieves the port number on which the codec sends the IP beacon when the `amx-enable` target is set to *on*. When used with the `set` verb, this target sets the port number on which the device sends the IP beacon when the `amx-enable` target is set to *on*. The default is 9131.

get Arguments:

None

get Examples:

```
get network amx-port
9131

ok,00

get network amx-port -V
AMX Master Port
9131

ok
```

set Arguments:

`<port number>`	Specify the port number on which the codec sends the IP beacon message when the `amx-enable` target is set to *on*.

set Examples:

```
set network amx-port 9131

ok,00
```

Commit the change:

```
set network commit

ok,00
```

amx-revision

The **amx-revision** target retrieves the revision number of the AMX interface that supports the codec. This target applies to the **get** verb.

get Arguments:

   None

get Examples:

```
get network amx-revision
1.0.0

ok,00

get network amx-revision -V
AMX Beacon Revision
1.0.0

ok
```

commit

The **commit** target commits the network settings. Network settings that are changed but not committed do not take effect until the next system reboot. This target applies to the **set** verb.

> **Note:** If you change network settings using the following commands and then commit the changes with the **set network commit** command, the system reboots:

- **set network ipv6 manual -i** *address*
- **set network ipv6 auto**
- **set network vlan id** *value*
- **set network reserved-ports -T** *port*

   > **Note:** Issuing the **set network commit** command restarts the system only if the change impacts ports already in use.

- **set network reserved-ports -t** *port*

Arguments:

   None

Examples:

   set network commit

   ok,00

dns

When used with the **get** verb, the **dns** target retrieves the current Directory Name Service settings. When used with the **set** verb, this target configures the Directory Name Service settings to allow the use of named hosts instead of IP addresses.

get Arguments:

> None

get Examples:

> **get network dns**
> *10.10.10.1,10.10.10.2,10.10.10.3,example.com*
>
> ok,00
>
> **get network dns -V**
> Primary DNS   Secondary DNS   Tertiary DNS   Domain Search List
> *10.10.10.1   10.10.10.2   example.com*
>
> ok

set Arguments:

[-i *ipaddress*]	Specify an IP address for a DNS server. Up to 3 servers may be specified.
[-r]	Reset the DNS servers and search domains instead of appending additional servers/domains.
[-s *domain*]	Specify a search domain (used for unqualified hostname resolution).

set Examples:

> **set network dns -i *10.10.11.1* -i *10.10.11.2* -s *ls.com* -s *cc.com***
>
> ok,00

Commit the change:

> **set network commit**
>
> ok,00

hostname

The `hostname` target sets the network hostname for the system. Use a name that is similar or the same as the system name to avoid confusion. If DHCP is used for the network configuration, the hostname will be published to the DHCP server allowing name based lookups for the system. This target applies to the `set` verb.

Arguments:

<hostname>	Specify the hostname for the system.

Examples:

> **set network hostname *lifesize-passport***
>
> ok,00

Commit the change:

> **set network commit**
>
> ok,00

ipv4

When used with the `get` verb, the `ipv4` target retrieves the current Internet Protocol Version 4 network configuration.

When used with the `set` verb, this target uses the `dhcp` and `static` targets to configure Internet Protocol Version 4 (ipv4) network parameters. The `dhcp` target configures the network to use Dynamic Host Control Protocol for the network settings. The `static` target controls the configuration of the network interface when you specify a static IP address for the device.

get Arguments:

> None

get Examples:

> **get network ipv4**
> *dhcp,10.10.15.166,255.255.255.0,10.10.15.255,10.10.15.1,*
> *   00:13:fa:00:24:a1,jsmith-ls*
>
> ok,00

```
get network ipv4 –V
```

```
Mode  IP Address     Network Mask   Broadcast IP  Gateway IP   MAC Address       Hostname

dhcp  10.10.15.166   255.255.255.0  10.10.15.255  10.10.15.1   00:13:fa:00:24:a1  jsmith-ls
```

```
ok
```

set Arguments (DHCP target):

None

set Examples (DHCP target):

```
set network ipv4 dhcp
```

```
ok,00
```

Commit the change:

```
set network commit
```

```
ok,00
```

set Arguments (static target):

`[-i ipaddr]`	Specify the IP address of the device.
`[-n netmask]`	Specify the network mask that defines the extent of the local network.
`[-g gateway]`	Specify the gateway address for routing traffic outside of the network defined by the IP address and network mask. The gateway device must be within the network.

set Examples (static target):

```
set network ipv4 static -i 10.10.11.12 -n 255.255.0.0 -g
   10.10.1.1
```

```
ok,00
```

Commit the change:

```
set network commit
```

```
ok,00
```

ipv6

When used with the **get** verb, the **ipv6** target retrieves the current Internet Protocol Version 6 network configuration.

When used with the **set** verb, this target uses the following targets to configure Internet Protocol Version 6 (IPv6) networking parameters:

- **auto** enables the system to determine the networking parameters from the network without further user intervention.

- **manual** enables you to enter the IPv6 addresses of the system and the router manually

    The **manual** .

- **off** disables IPv6 networking.

get Arguments:

    None

get Examples:

```
get network ipv6
yes,auto,yes,ipv6Address,

ok,00
get network ipv6 -V
Enabled    Mode   Active   IP Address        Router
yes        auto   yes      ipv6Address

ok
```

set Arguments (auto target):

    None

set Examples (auto target):

```
set network ipv6 auto

ok,00
```

Commit the change (causes a system reboot):

```
set network commit

ok,00
```

set Arguments (manual target):

`[-i ipaddress]`	Specify the IPV6 address for the system.
`[-r routerip]`	Specify the IPV6 address of the router.

set Examples (manual target):

set network ipv6 manual -i *address*

ok,00

Commit the change (causes a system reboot):

set network commit

ok,00

set Arguments (off target):

None

set Examples (off target):

set network ipv6 off

ok,00

Commit the change (causes a system reboot):

set network commit

ok,00

nat

When used with the **get** verb, the **nat** target retrieves the Network Address Translation (NAT) settings for the system.

When used with the **set** verb, this target uses the **disabled** and **enabled** targets to configure NAT. The **disabled** target disables the use of NAT traversal on the device. The **enabled** target enables the use of NAT traversal on the device.

get Arguments:

None

network

get Examples:

get network nat

disabled

ok,00

get network nat -V
Static NAT   Public IP
enabled   10.10.11.111

ok

set Arguments (disabled target):

None

set Examples (disabled target):

set network nat disabled

ok,00

Commit the change:

set network commit

ok,00

set Arguments (enabled target):

<ipaddress>	Specify the public IP address of the LifeSize device.

set Examples (enabled target):

set network nat enabled *ipaddress*

ok,00

Commit the change:

set network commit

ok,00

ntp-server

When used with the **get** verb, the **ntp-server** target retrieves the address of the current Network Time Protocol server. When used with the **set** verb, this target specifies the Network Time Protocol server to use to keep the system clock synchronized with a common time source.

get Arguments:

　　None

get Examples:

```
get network ntp-server
10.10.11.10

ok,00

get network ntp-server -V
NTP Server
10.10.11.10

ok
```

set Arguments:

<ntpserver>	Specify the IP address of the NTP server.

set Examples:

```
set network ntp-server 10.10.11.10

ok,00
```

Commit the change:

```
set network commit

ok,00
```

qos

When used with the **get** verb, the **qos** target retrieves the configuration of the network Quality of Service (QoS) options for the system.

When used with the **set** verb, this target uses the following targets to configure the QoS options for the system:

- **diffserv** configures the network QoS for DiffServ.
- **intserv** configures the network QoS for IntServ (IP Precedence).
- **none** disables network QoS.

Arguments:

    None

Examples:

    get network qos
    DiffServ,46,34,46,

    ok,00

    get network qos -V

QoS Mode	Audio Priority	Video Priority	Data Priority	Type Of Service
IntServ	*5*	*4*	*5*	*Minimize Cost*

    ok

set Arguments (diffserv target):

`[-a {0..63}]`	Specify the audio packet priority.
`[-d {0..63}]`	Specify the data packet priority.
`[-v {0..63}]`	Specify the video packet priority.

set Examples (diffserv target):

    set network qos diffserv -a *46* -d *46* -v *34*

    ok,00

Commit the change:

    set network commit

    ok,00

set Arguments (intserv target):

`[-a {0..7}]`	Specify the audio packet priority.
`[-d {0..7}]`	Specify the data packet priority.
`[-v {0..7}]`	Specify the video packet priority.
`[-t {none|min-delay| min-cost| max-rely|max-thru}]`	Specify the type of service used by your network: None, Minimize Delays, Minimize Cost, Maximize Reliability, or Maximize Throughput

set Examples (intserv target):

set network qos intserv -a *6* -d *4* -v *3* -t *min-delay*

ok,00

Commit the change:

set network commit

ok,00

set Arguments (none target):

None

set Examples (none target):

set network qos none

ok,00

Commit the change:

set network commit

ok,00

reserved-ports

When used with the **get** verb, the **reserved-ports** target retrieves the configuration of ports reserved for use by the device. When used with the **set** verb, this target specifies the upper and lower bounds for the ports reserved for use by the device.

get Arguments:

   None

get Examples:

   get network reserved-ports
   64000,64999,64000,64999

   ok,00

   get network reserved-ports -V
   UDP Low Port   UDP High Port   TCP Low Port   TCP High Port
   64000   *64999*   *64000*   *64999*

   ok

set Arguments:

[-T {1025..65535}]	Specify the upper bound for TCP reserved ports
[-U {1025..65535}]	Specify the upper bound for UDP reserved ports
[-t {1025..65535}]	Specify the lower bound for TCP reserved ports
[-u {1025..65535}]	Specify the lower bound for UDP reserved ports

set Examples:

   set network reserved-ports -t *30000* **-T** *40000*

   ok,00

Commit the change:

   Note: Committing the change causes a system reboot if the -t argument was used. If the -T argument was used, committing the change causes a system reboot only if the change impacts ports already in use.

   set network commit

   ok,00

speed

When used with the **get** verb, the **speed** target shows the actual speed of the network port. Possible values are `100-fd` (100Mbps, full duplex), `100-hd` (100Mbps, half duplex), `10-fd` (10Mbps, full duplex) and `10-hd` (10Mbps, half duplex). When used with the **set** verb, this target configures the network port default speed.

get Arguments:

   None

get Examples:

```
get network speed
100-fd

ok,00

get network speed -V
Network Speed
10-fd

ok,00
```

set Arguments:

`<{auto|`
`100-auto|`
`100-fd|`
`10-auto|`
`10-fd}>`
	Specify the network speed. `Auto` negotiates 10 or 100Mbps and duplex. `100-auto` and `10-auto` negotiate only duplex. `100-fd` and `10-fd` do not negotiate. Set the speed to `auto` unless the remote networking equipment is incapable of auto negotiation.

set Examples:

```
set network speed auto

ok,00
```

Commit the change:

```
set network commit

ok,00
```

network

status

The `status` target shows the current status of the network connection. This target applies to the `get` verb. Possible values include:

- `connected`
- `binding` (attempting to retrieve IP configuration)
- `no dhcp response` (DHCP server timed out)
- `unconnected`

Arguments:

　　None

Examples:

```
get network status
```
connected

```
ok,00
```

```
get network status -V
Network State
```
binding

```
ok
```

transit

The **transit** object controls the LifeSize Transit parameters.

ice

When used with the **get** verb, the **ice** target retrieves the state of the Interactive Connectivity Establishment feature when placing calls with LifeSize Transit. When used with the **set** verb, this target enables or disables the Interactive Connectivity Establishment feature when placing calls with LifeSize Transit.

get Arguments:

    None

get Examples:

```
get network transit ice
on

ok,00

get network transit ice -V
State
off

ok
```

set Arguments:

`<{on|off}>`	Enable or disable the feature.

set Examples:

```
set network transit ice on

ok,00
```

Commit the change:

```
set network commit

ok,00
```

server

When used with the **get** verb, the **server** target configures the LifeSize Transit or STUN/TURN server parameters. When used with the **set** verb, this target configures the LifeSize Transit or STUN/TURN server parameters.

get Arguments:

None

get Examples:

get network transit server
server1 server2,transitUser

ok,00

get network transit server -V
Hostname   Username
server1 server2 transitUser

ok

set Arguments:

[-a]	Append the -i parameters to the end of the existing list (default is to replace the list).
[-i *server*]	Specify the server hostname or IP to use (may be repeated).
[-p *password*]	Specify the password to use to log into the server.
[-u *username*]	Specify the username to use to log into the server.

set Examples:

set network transit server -i *server1* -i *server2* -u *user* -p *pass*

ok,00

Commit the change:

set network commit

ok,00

service

When used with the **get** verb, the **service** target displays the state of the LifeSize Transit feature. When used with the **set** verb, this target enables or disables the LifeSize Transit feature.

get Arguments:

None

get Examples:

get network transit service
on

ok,00

get network transit service -V
State
off

ok

set Arguments:

`<{on|off}>`	Enable or disable the feature.

set Examples:

set network transit service *on*

ok,00

Commit the change:

set network commit

ok,00

network

signaling

When used with the **get** verb, the **signaling** target retrieves the signaling mode for the LifeSize Transit feature. When used with the **set** verb, this target configures the signaling mode for the LifeSize Transit feature.

get Arguments:

    None

get Examples:

```
get network transit signaling
udp-tcp

ok,00

get network transit signaling -V
Mode
tcp

ok
```

set Arguments:

`<{udp-tcp|tcp}>`	Automatically choose between UDP and TCP protocols, or force to use TCP only.

set Examples:

```
set network transit signaling udp-tcp

ok,00
```

Commit the change:

```
set network commit

ok,00
```

web

When used with the **get** verb, the **web** target retrieves the LifeSize Transit web proxy parameters. When used with the **set** verb, this target configures the LifeSize Transit web proxy parameters.

get Arguments:

   None

get Examples:

   get network transit web
   http://webproxy.com/lifesize,transitUser

   ok,00

   get network transit web -V
   Proxy URL   Username
   http://webproxy.com/lifesize   transitUser

   ok

set Arguments:

[-U *url*]	Specify the URL of the web proxy service.
[-p *password*]	Specify the password to use to log into the server.
[-u *username*]	Specify the username to use to log into the server.

set Examples:

   set network transit web -U *http://webproxy.com* -u *user* -p *pass*

   ok,00

Commit the change:

   set network commit

   ok,00

network

vlan

The **vlan** object controls VLAN configuration parameters.

id

When used with the **get** verb, the **id** target retrieves the current VLAN identifier of the static VLAN to which the system is assigned. When used with the **set** verb, this target specifies the VLAN identifier of the static VLAN to which the system is assigned. If you specify the **vlan id**, the LifeSize system applies a VLAN tag to outgoing packets and only accepts incoming tagged packets that have the same VLAN identifier.

get Arguments:

    None

get Examples:

```
get network vlan id
10

ok,00

get network vlan id -V
ID
10
```

set Arguments:

`<{1..4094}>`	Specify the VLAN identifier of the static VLAN to which the system is assigned.

set Examples:

```
set network vlan id 15

ok,00
```

Commit the change:

```
set network commit

ok,00
```

password

The **password** target changes the user's password while running the CLI. This target applies to the **set** verb. When used with the optional arguments, **password** behaves like other CLI targets. However, using the arguments is insecure: the **set password** command may show up in the saved history for the CLI across login sessions, allowing disclosure of the new password. Also, checks for minimum length and complexity are not performed on the password in this mode. For security reasons, LifeSize recommends that you use the interactive version.

Note: The interactive mode of this command does not conform to the standard output specification, because it uses the standard passwd utility to perform the change.

Arguments:

[*old-password*]	Specify the current password for the CLI.
[*new-password*]	Specify the new password for the CLI.

Examples:

Interactive:

set password
```
Changing password for auto
Old password:
Enter the new password (minimum of 5, maximum of 127 characters)
Please use a combination of upper and lower case letters and num-
    bers.
New password:
Re-enter password:
Password changed.

ok,00
```

Non-interactive:

set password *lifesize 123ABC!@#abc*

```
ok,00
```

prompt

The **prompt** target changes the default prompt ($) to any user specified string. This target applies to the **set** verb.

Arguments:

<string>	Specify the new prompt "string "; use "" for an empty prompt.

Examples:

```
set prompt "% "

ok,00
% set prompt "-> "

ok,00
->
```

redial-list

The **redial-list** target retrieves the redial call list. This target applies to the **get** verb.

Arguments:

None

Examples:

```
get redial-list
1,Sunbob2,10.10.11.116,10.10.11.116,Video,Outgoing,Yes,auto,auto
2,10.10.11.186,10.10.11.186,10.10.11.186,Video,Manual,No,
   auto,512
3,Sunbob2,10.10.11.116,10.10.11.116,Audio,Incoming,Yes,h323,auto
4,10.10.11.186,10.10.11.186,10.10.11.186,Audio,
   Multiway,No,auto,auto
5,10.10.11.155,10.10.11.155,10.10.11.155,Audio,
   Outgoing,No,auto,auto

ok,00
```

```
get redial-list -V
```

Index	Name	Number	IP Address	Type	Origin	Locked	Protocol	Bandwidth
1	Sunbob2	10.10.11.116	10.10.11.116	Video	Outgoing	Yes	auto	auto
2	10.10.11.186	10.10.11.186	10.10.11.186	Video	Manual	No	auto	512
3	Sunbob2	10.10.11.116	10.10.11.116	Audio	Incoming	Yes	h323	auto
4	10.10.11.186	10.10.11.186	10.10.11.186	Audio	Multiway	No	auto	auto
5	10.10.11.155	10.10.11.155	10.10.11.155	Audio	Outgoing	No	auto	auto

```
ok
```

The valid values for the `Type` column are *Audio*, *Video*, *Multiway*, and *Unknown*. The values for the `Origin` column are *Manual*, *Outgoing*, *Incoming*, *Multiway*, and *Unknown,* where `Origin` refers to how the entry was placed into the redial list. Entries that are locked cannot be removed from the redial list with new entries. The protocol values are the same as for the **`control call dial`** command's `-p` argument, and the bandwidths are the same as for the `-b` argument.

serial

The **`serial`** object allows configuration of a device connected to the USB serial port through the **`port3`** target. When used with the **`get`** verb, the **`port3`** target retrieves the configuration settings for the specified USB port. When used with the **`set`** verb, this target configures available settings for the specified USB port.

get Arguments:

> None

get Examples:

```
get serial port3
38400,none,none,backspace,

ok,00
```

```
get serial port3 -V
```

Speed	Flow Control	Shell	Erase Key	VISCA Input
38400	none	none	backspace	

```
ok,00
```

serial

set Arguments:

[-b {1200|2400| 4800|9600|19200| 38400|57600| 115200}]	Specify the speed of the serial port.
[-e {backspace| delete}]	Specify the erase character to use.
[-f {hardware| software|none}]	Specify the flow control method to use. Hardware flow control does not function on LifeSize video communications systems. Set flow control to software or none. Choosing hardware is the same as choosing none.
[-s {auto|none}]	Specify the shell to run on the serial port. The auto argument uses the shell you are currently running; none disables the serial port.

set Examples:

set serial port3 -b *115200* -e *backspace* -f *software* -s *auto*

ok,00

sip

The **sip** object control Session Initiation Protocol (SIP) configuration settings. If you use the **set** verb with a command that contains the **sip** object, you must issue the **set sip commit** command to commit the change. The **commit** target commits the SIP settings. SIP settings that are changed but not committed do not take effect until the next system reboot.

Note: When followed by the **set sip commit** command. some sip commands cause the system to restart. For a list of these commands, refer to "commit" on page 130.

The following targets apply to the **sip** object.

authorization

When used with the **get** verb, the **authorization** target retrieves the username for authorization with the SIP registrar. For security reasons, the associated password is not displayed. When used with the **set** verb, this target configures the username used for authorization with the SIP registrar.

get Arguments:

    None

get Examples:

    get sip authorization
    sipuser

    ok,00

    get sip authorization -V
    Username
    sipuser

    ok

set Arguments:

username	Specify the username used for authorization.
password	Specify the password used for authorization.

set Examples:

    set sip authorization *sipuser sippassword*

    ok,00

Commit the change:

set sip commit

ok,00

commit

The **commit** target commits the SIP settings. Any SIP settings that are changed but not committed do not take effect until the next system reboot. This target applies to the **set** verb.

Note: If you change SIP settings using the following commands and then commit the changes with the **set sip commit** command, the system reboots:

- **set sip tcp**
- **set sip tls**
- **set sip udp**

Arguments:

None

Example:

set sip commit

ok.00

proxy

When used with the **get** verb, the **proxy** target retrieves the SIP proxy settings. When used with the **set** verb, this target sets the SIP proxy configuration.

get Arguments:

None

get Examples:

get sip proxy
disabled,proxy.example.com,5060

ok,00

```
get sip proxy -V
```

```
State             IP Address          Port
```

```
enabled          proxy.example.com    5060
```

```
ok
```

set Arguments:

`<{enabled|disabled}>`	Enables or disables the use of the SIP proxy.
`[ip]`	Set the IP address or hostname of the SIP proxy. Only valid when enabled is chosen.
`[port]`	Specify the port to use on the proxy. The default is `5060` or the previously set value. Only valid when `enabled` is chosen.

set Examples:

```
set sip proxy enabled proxy.example.com
```

```
ok,00
```

```
set sip proxy disabled
```

```
ok,00
```

```
set sip proxy enabled proxy.sip.com 6060
```

```
ok,00
```

Commit the change:

```
set sip commit
```

```
ok.00
```

register

When used with the **get** verb, the **register** target shows the current registration status for SIP. When used with the **set** verb, this target registers the device with the configured SIP server or proxy. Use this command only after completing all other SIP configuration tasks. Since registration may take an arbitrarily long time, this command returns immediately. Use the **get sip register** command to retrieve the registration status.

get Arguments:

    None

get Examples:

    get sip register
    unregistered

    ok,00

    get sip register -V
    Status
    registered

    ok

set Arguments:

    None

set Examples:

    set sip register

    ok,00

Commit the change:

    set sip commit

    ok.00

registrar

When used with the **get** verb, the **registrar** target retrieves the current SIP registrar settings. When used with the **set** verb, this target configures the SIP registrar settings.

get Arguments:

   None

get Examples:

   get sip registrar
   disabled,sip.example.com,5060

   ok,00

   get sip registrar -V
   State   IP Address   Port
   enabled   sip.example.com   5060

   ok

set Arguments:

`<{enabled|disabled}>`	Enables or disables the use of the SIP registrar.
`[ip]`	Set the IP address or hostname of the SIP registrar. Only valid when `enabled` is chosen.
`[port]`	Optional: Specify the port to use on the registrar. The default is 5060 or the previously set value. Only valid when `enabled` is chosen.

set Examples:

   set sip registrar *enabled sip.example.com*

   ok,00

   set sip registrar *disabled*

   ok,00

   set sip registrar *enabled registrar.sip.com 6060*

   ok,00

sip

Commit the change:

set sip commit

ok.00

server-type

When used with the **get** verb, the **server-type** target shows the type of SIP server used with SIP calls. When used with the **set** verb, this target specifies the type of SIP server to use with SIP calls.

get Arguments:

None

get Examples:

get sip server-type
generic

ok,00

get sip server-type -V
server type
generic

ok

set Arguments:

`<{generic|ocs}>`	Specify `ocs` if you are using Microsoft Office Communications Server.

set Examples:

set sip server-type *ocs*

ok,00

Commit the change:

set sip commit

ok.00

sip

When used with the **get** verb, the **sip** target shows whether SIP calls are enabled or disabled. When used with the **set** verb, this target controls whether SIP calls are enabled or disabled.

get Arguments:

    None

get Examples:

get sip sip
enabled

ok,00

get sip sip -V
Sip
enabled

ok

set Arguments:

<{enabled|disabled}>	Specify whether to enable or disable SIP calls.

set Examples:

set sip sip *disabled*

ok,00

Commit the change:

set sip commit

ok.00

tcp

When used with the **get** verb, the **tcp** target shows the configuration of the TCP options for SIP calls. When used with the **set** verb, this target configures the TCP options for SIP calls. If configuring the device for SIP calls, either the **tcp** or **udp** target must be enabled.

get Arguments:

> None

get Examples:

> **get sip tcp**
> *enabled,5060*
>
> ok,00
>
> **get sip tcp -V**
> State   Port
> *disabled   5060*
>
> ok

set Arguments:

`<{enabled|` `disabled}>`	Enables or disables the use of TCP for SIP calls.
`[port]`	Specify the port to use for SIP calls. The default is `5060` or the previously set value. Only valid when `enabled` is chosen.

set Examples:

> **set sip tcp *enabled***
>
> ok,00
>
> **set sip tcp *disabled***
>
> ok,00
>
> **set sip tcp *enabled 5060***
>
> ok,00

Commit the change:

    set sip commit

    ok.00

tls

When used with the **get** verb, the **tls** target shows whether TLS signaling for use with SIP calls is enabled or disabled and the port number that is used.

get Arguments:

    None

get Examples:

    get sip tls
    disabled,5061

    ok,00

    get sip tls -V
    State   Port
    disabled   5061

    ok

set Arguments:

`<{enabled|disabled}>`	Enables or disables the use of TLS signaling with SIP calls.
`[port]`	Optional: Specify the port to use when TLS signaling is enabled. The default is `5061` or the previously set value. Only valid when `enabled` is chosen.

set Examples:

    set sip tls enabled *5062*

    ok,00

Commit the change:

    set sip commit

    ok.00

sip

tls-cert

When used with the **get** verb, the **tls-cert** target retrieves a CA certificate installed on the system for validating the certificate sent by the SIP registrar/proxy when SIP registrar/proxy validation is enabled with the set **sip tls-server-validate** command. When used with the set verb, this target adds a CA certificate to the device.

get Arguments:

   None

get Examples:

```
get sip tls-cert
certificate data

ok,00

get sip tls-cert -V
TLS Certificate
certificate data

ok
```

set Arguments:

   None

set Examples:

```
set sip tls-cert << EOF
certificate data
EOF

ok,00
```

Commit the change:

```
set sip commit

ok.00
```

tls-server-validate

When used with the **get** verb, the **tls-server-validate** target shows whether SIP registrar/proxy server validation is enabled or disabled. When used with the **set** verb, this target controls whether SIP registrar/proxy server validation is enabled or disabled. If you enabled TLS signaling on the LifeSize system for SIP calls, you can use this command and the **set sip tls-cert** command to confirm the identity of the SIP Registrar/Proxy.

get Arguments:

> None

get Examples:

> **get sip tls-server-validate**
> *disabled*
>
> ok,00
>
> **get sip tls-server-validate -V**
> TLS Server Validation
> *disabled*
>
> ok

set Arguments:

<{enabled|disabled}>	Specify whether to enable or disable SIP registrar/proxy server validation.

set Examples:

> **set sip tls-server-validate *enabled***
>
> ok,00

Commit the change:

> **set sip commit**
>
> ok.00

udp

When used with the **get** verb, the **udp** target retrieves the configuration of the UDP options for SIP calls. When used with the **set** verb, this target configures the UDP options for SIP calls. If configuring the device for SIP calls, either the **tcp** or **udp** target must be enabled.

get Arguments:

　　None

get Examples:

```
get sip udp
disabled,5060

ok,00

get sip udp -V
State          Port
enabled        5060

ok
```

set Arguments:

`<{enabled|disabled}>`	Enables or disables the use of UDP for SIP calls.
`[port]`	Optional: Specify the port to use for SIP calls. The default is 5060 or the previously set value. Only valid when enabled is chosen.

set Examples:

```
set sip udp enabled

ok,00

set sip udp disabled

ok,00

set sip udp enabled 7000

ok,00
```

Commit the change:

> **set sip commit**

> ok.00

username

When used with the **get** verb, the **username** target retrieves the current SIP username. When used with the **set** verb, this target sets the SIP username for the system.

get Arguments:

> None

get Examples:

> **get sip username**
> *lifesize*

> ok,00

> **get sip username -V**
> Name
> *lifesize*

> ok

set Arguments:

<name>	Set the username for the system.

set Examples:

> **set sip username *lifesize***

> ok,00

Commit the change:

> **set sip commit**

> ok.00

via-proxy

When used with the **get** verb, the **via-proxy** target shows whether SIP registration uses a proxy to connect to the registrar, or connects directly. When used with the **set** verb, target target controls whether SIP registration uses a proxy to connect to the registrar or connects directly.

get Arguments:

> None

get Examples:

> **get sip via-proxy**
> *proxy*
>
> ok,00
>
> **get sip via-proxy -V**
> State
> *direct*
>
> ok

set Arguments:

`<{direct|proxy}>`	Choose direct connection to the registrar or the proxy.

set Examples:

> **set sip via-proxy *direct***
>
> ok,00
>
> **set sip via-proxy *proxy***
>
> ok,00

Commit the change:

> **set sip commit**
>
> ok.00

skype

The following targets apply to the **skype** configuration object.

allow-from

When used with the **get** verb, the **allow-from** target shows the class of user from which incoming calls are allowed. When used with the **set** verb, this target sets the class of user from which incoming calls are allowed.

get Arguments:

None

get Examples:

get skype allow-from
contacts

ok,00

get skype allow-from -V
Value
contacts

ok

set Arguments:

`<{anyone/contacts}>`	You can choose to allow Skype calls from any user or only accept Skype calls from your contact list.

set Examples:

set skype allow-from anyone

ok,00

auto-logout

When used with the **get** verb, the **auto-logout** target returns the Skype auto logout setting. When used with the **set** verb, this target sets the Skype auto-logout feature from `never` to `1400` minutes before the system automatically logs out the current user.

get Arguments:

   None

get Examples:

   get skype auto-logout
   never

   ok,00

   get skype auto-logout -V
   Value
   60

   ok

set Arguments:

`<{never/1|5|10|15|20|30` `|60|120|180|240|480|600` `|720|960|1200|1400}>`	Select how long in minutes before the system automatically logs out the current Skype user.

set Examples:

   set skype auto-logout *60*

   ok,00

login

When used with the **get** verb, the **login** target retrieves the username last used to log in to Skype as well as the login status. Login status can return the following: *Undefined*, *Logged In*, *Logged Out*, *Server Failure*, *Invalid Skype Name*, *Invalid Password*, and *Connection Failure*. For security reasons, the associated password is not displayed. When used with the **set** verb, this target configures the username and password used to log in to Skype.

get Arguments:

   None

get Examples:

> **get skype login**
> *skypeuser,Logged In*
>
> ok,00
>
> **get skype login -V**
> Username   Status
> *skypeuser Logged Out*
>
> ok

set Arguments:

<username>	Specify the username used for authorization.
<password>	Specify the password used for authorization.

set Examples:

> **set skype login *skypeuser skypepassword***
>
> ok,00

logout

The **logout** target logs out of Skype. This target applies to the **set** verb.

set Arguments:

> None

set Examples:

> **set skype logout**
>
> ok,00

snmp

The following targets apply to the **snmp** server configuration object.

contact

When used with the **get** verb, the **contact** target retrieves the SNMP contact name. When used with the **set** verb, this target sets the SNMP contact name on the SNMP server running on the device.

get Arguments:

   None

get Examples:

```
get snmp contact
Administrator

ok,00

get snmp contact -V
SNMP Contact
Administrator

ok
```

set Arguments:

<contactname>	Specify the contact name for the SNMP server. If the contact name contains more than one word separated by a space, enclose the name in quotes (" ").

set Examples:

```
set snmp contact Administrator

ok,00
```

description

The **description** target retrieves the SNMP description of the system. This target applies to the **get** verb.

Arguments:

   None

Examples:

> **get snmp description**
> *LifeSize Passport*
>
> ok,00
>
> **get snmp description -V**
> SNMP System Description
> *LifeSize Passport*
>
> ok

enable

When used with the **get** verb, the **enable** target shows whether or not the SNMP service is enabled. When used with the **set** verb, this target enables or disables the SNMP service.

get Arguments:

> None

get Examples:

> **get snmp enable**
> *on*
>
> ok,00
>
> **get snmp enable -V**
> Value
> *off*
>
> ok

set Arguments:

<{on|off}>	Enable or disable the SNMP service.

set Examples:

> **set snmp enable *on***
>
> ok,00

location

When used with the **get** verb, the **location** target shows the configured location for the SNMP service. When used with the **set** verb, this target sets the configured location for the SNMP service.

get Arguments:

    None

get Examples:

    get snmp location
    Austin

    ok,00

    get snmp location -V
    SNMP Location
    Austin

    ok

set Arguments:

<location>	Specify the location for the SNMP service. If the location contains more than one word separated by a space, enclose the location in quotes (" ").

set Examples:

    set snmp location *Austin*

    ok,00

system-name

The `system-name` target retrieves the SNMP system name. This target applies to the `get` verb.

get Arguments:

> None

get Examples:

> **get snmp system-name**
> *foo*
>
> ok,00
>
> **get snmp system-name -V**
> SNMP System Name
> *foo*
>
> ok

user

When used with the `get` verb, the *user* target retrieves the SNMP usernames. When used with the `set` verb, this target adds or deletes SNMP users.

get Arguments:

> None

get Examples:

> **get snmp user**
> *user1*
> *Control*
> *user2*
>
> ok,00

snmp

```
get snmp user -V
Username
user1
Control
user2

ok
```
Note: The Control user is a default user for use with LifeSize Control. You can delete this user if you are not using LifeSize Control or delete this user and create a different user for use with LifeSize Control. Use the **set snmp user** command to delete and create SNMP users.

set Arguments:

-a	Add the specified user (cannot be used with -d).
-d	Delete the specified user (cannot be used with -a).
<username>	Specify the username. usernames must not contain spaces.
<password>	Specify the password for the user. Required with -a. The password must be at least 8 characters in length and must not contain spaces.

set Examples:

Add a user:

set snmp user -a *username password*

Delete a user:

set snmp user -d *username*

v3trapdestination

When used with the **get** verb, the **v3trapdestination** shows the current version 3 SNMP trap destinations (where SNMP traps are sent). When used with the **set** verb, this target adds or removes entries from the list of version 3 SNMP trap destinations.

Note: The user's password is not displayed.

get Arguments:

None

get Examples:

`get snmp v3trapdestination`
Control,10.10.11.12
joeuser,169.254.101.2

`ok,00`

`get snmp v3trapdestination -V`

Username	Host/IP Address
Control	*10.10.11.12*
joeuser	*169.254.101.2*

`ok`

set Arguments:

-a	Add the specified destination (cannot be used with -d).
-d	Delete the specified destination (cannot be used with -a).
<username>	Specify the username associated with the trap destination. usernames must not contain spaces.
<password>	Specify the password for the user. Passwords must not contain spaces. Password must be at least 8 characters in length.
<ipaddress>	Specify the IP address of the trap destination. IP addresses must not contain spaces.

Note: The *<username>*, *<password>*, and *<ipaddress>* arguments are required with the -a and -d arguments. Either -a or -d must be specified. Users specified with this command appear in the output of the **`get snmp user`** command.

set Examples:

`set snmp v3trapdestination -a `*`user1 password 10.10.11.10`*

`ok,00`

`set snmp v3trapdestination -d `*`user1 password 10.10.11.10`*

`ok,00`

version

The **version** target retrieves the SNMP version number for the SNMP server running on the device. This target applies to the **get** verb.

Arguments:

   None

Examples:

```
get snmp version
3

ok,00

get snmp version -V
SNMP Version
3

ok
```

ssh

The following targets apply to the **ssh** object.

keys

When used with the **get** verb, the **keys** target retrieves information about the currently installed ssh authorized keys. When used with the **set** verb, this target sets the ssh authorized keys for the **auto** user. Authorized keys allow the remote user to log into the system without using a password.

get Arguments:

   None

get Examples:

```
get ssh keys
ssh-rsa,user@lifesize.com

ok,00
```

```
get ssh keys -V
```

```
Type               Owner
```

```
ssh-rsa            user@lifesize.com
```

```
ok
```

set Arguments:

[-c]	Clear the keys file, removing all installed keys.
[-i]	Install a new ssh key. The key is read from standard input and must be less than 4096 characters in length. You may specify multiple keys on separate lines. A maximum of 64 keys are supported.
[-r *owner*]	Remove an existing ssh key. The owner field must match the prefix of the comment field in the key file, ignoring case (for example, an owner of "li" would match all comment fields beginning with "li" in any case).

Note: Either **-i** or **-r** must be specified.

set Examples:

Manually enter an ssh key using a here document or paste the key file into the command line:

```
set ssh keys -i << EOF
ssh-rsa key_string user@lifesize.com
ssh-rsa key2_string user2@lifesize.com
EOF
```

```
ok,00
```

Copy your own public key file to the auto user's authorized key file:

```
sh% cat ~/.ssh/id_rsa.pub | ssh auto@10.10.1.1 set ssh keys -i
```

```
ok,00
```
sh%

Remove the specified key:

```
set ssh keys -r user@lifesize.com
```

```
ok,00
```

Note: The key file must not have any text prior to the key data and the key type, value, and comment (owner) must be on a single line.

service

When used with the **get** verb, the **service** target shows whether the ssh service is enabled or disabled. When used with the **set** verb, this target enables or disables the ssh service. An ssh session in progress is not affected if the service is disabled.

Note: Be aware that if the device does not have a serial port and you disable the ssh service and then quit the session, you may need to use the user interface or web administration interface to re-enable the ssh service.

get Arguments:

   None

get Examples:

   get ssh service
   on

   ok,00

   get ssh service -V
   Secure Shell Service
   off

   ok

set Arguments:

`<{off|on}>`	Disable or enable the ssh service.

set Examples:

   set ssh service *on*

   ok,00

system

The **system** object allows setting of certain system-specific parameters, for example, the system name and may be useful for tracking and monitoring inventory. The following targets apply to the **system** object.

admcontrol

When used with the **get** verb, the **admcontrol** target shows the current setting of the admission control feature. If enabled, the device uses admission control to preserve useful bandwidth for existing call participants. When used with the **set** verb, this target controls whether the admission control is enabled for the device.

get Arguments:

   None

get Examples:

```
get system admcontrol
enabled

ok,00

get system admcontrol -V
Admission Control
enabled

ok
```

set Arguments:

`<{enabled|disabled}>`	Specify *enabled* to enable admission control.

set Examples:

```
set system admcontrol enabled

ok,00
```

autoreboot

When used with the **get** verb, the **autoreboot** target shows the current setting of the nightly automatic reboot feature. If enabled, the device reboots nightly if the system is idle. When used with the **set** verb, this target controls whether or not the device automatically reboots each evening.

get Arguments:

   None

get Examples:

```
get system autoreboot
off

ok,00

get system autoreboot -V
Nightly Reboot
on

ok
```

set Arguments:

`<{on|off}>`	Specify *on* to enable the reboot feature.

set Examples:

```
set system autoreboot on

ok,00
```

branding

When used with the **get** verb, the **branding** target retrieves the state of the logo branding feature. If set to **none**, no logo appears in the main screen of the user interface or in the logo screen saver. When used with the **set** verb, this target controls whether or not the company logo appears on the main screen of the user interface and in the logo screen saver.

get Arguments:

   None

get Examples:

```
get system branding
none

ok,00

get system branding -V
Company Logo
default

ok
```

set Arguments:

<{none|default}>	Specify none to disable the logo branding. Specify default to use the default logo.

set Examples:

```
set system branding none

ok,00
```

system

clean

The **clean** target removes personally identifiable information from the system, including call history logs, directory entries, system identity data, IP addresses, and Redial list entries. Use the **clean** target, for example, when you wish to use a system for customer demonstrations or for other uses that require the removal of personally identifiable information. This target applies to the **set** verb.

Note: Call history logs generated with the **-X** argument (**status call history -X**) are not cleaned with this target.

Arguments:

[-C]	Clean the call history logs (status call history).
[-a]	Clean all data.
[-c]	Clean the corporate directory, disable LDAP and Auto Discovery (get directory corporate, get directory ldap, get directory auto).
[-d]	Clean all directories (equivalent to -c -l -m).
[-i]	Clean system identity data (get system name, get system number, get system video-number, get network hostname).
[-l]	Clean the local directory (get directory local).
[-m]	Clean the meetings directory (get directory meeting).
[-n]	Clean the network config (get network ipv4, get network ipv6). Network is set to IPv4 static with no address and IPv6 is disabled. This occurs on reboot.
[-r]	Clean the redial list (get redial-list).

Examples:

Clean everything:

set system clean -a

ok,00

Clean only the directories and redial list:

set system clean -d -r

ok,00

corporate-dir-access

When used with the **get** verb, the **corporate-dir-access** target shows whether user access to the corporate directory is *enabled* or *disabled*. When used with the **set** verb, this target controls whether user access to the corporate directory is *enabled* or *disabled*. When set to ***disabled***, users cannot access the corporate directory.

get Arguments:

　　None

get Examples:

```
get system corporate-dir-access
enabled

ok,00
get system corporate-dir-access -V
Corporate Dir Access
enabled

ok
```

set Arguments:

<{enabled|disabled}>	Select disabled to restrict user access to the corporate directory.

set Examples:

```
set system corporate-dir-access disabled

ok,00
```

date

When used with the **get** verb, the **date** target shows the current system date and time in either the local time zone or as UTC time.

When used with the **set** verb, this target changes the system time and date. The value is always specified in terms of the local time zone.

get Arguments:

[-u]	Show the time in UTC instead of the local time zone.

system

get Examples:

```
get system date
2009,10,8,16,58,25

ok,00

get system date -u -V
Year      Month   Day     Hour      Minute      Second
2009      10      8       21        58          25

ok
```

set Arguments:

`[-H {0..23}]`	Specify the hour.
`[-M {0..59}]`	Specify the minute.
`[-S {0..59}]`	Specify the second.
`[-d {1..31}]`	Specify the day of month. February 31st is interpreted as March 2nd or 3rd depending on whether the year is a leap year or not.
`[-m {1..12}]`	Specify the month.
`[-y {2005..2025}]`	Specify the year.

set Examples:

```
# change only the time, not the day
set system date -H 4 -M 3 -S 0

ok,00

# change only the day of month
set system date -d 12

ok,00
```

do-not-disturb

When used with the **get** verb, the **do-not-disturb** target shows whether the system do not disturb preference is *enabled* or *disabled*. Select **enabled** with the **set** verb to prevent incoming calls. Select **disabled** to allow incoming calls.

Note: As soon as you issue the enable command, system do not disturb is activated whether the system is in a call or not.

get Arguments:

   None

get Examples:

```
get system do-not-disturb
enabled

ok,00
get system do-not-disturb -V
System Do Not Disturb
disabled

ok
```

set Arguments:

<{enabled|disabled}>	Select **disabled** to allow in coming call. Select **enabled** to block in coming calls.

set Examples:

```
set system do-not-disturb enabled

ok,00
```

fans

The **fans** target shows the current speed of all system fans. The speed shown is not recorded in a standard unit (such as revolutions per minute). The faster the fan spins, the higher the value reported. Systems with multiple fans report multiple values. This target applies to the **get** verb.

Arguments:

    None

Examples:

```
get system fans
655

ok,00

get system fans -V
Fan 1
639

ok
```

licensekey

When used with the **get** verb, the **licensekey** target retrieves the current license key installed on the system for upgrades. When used with the **set** verb, this target installs a license key or removes all license keys of a specified type.

get Arguments:

<-t maint>	Specify the type of license key.

get Examples:

```
get system licensekey -t maint
...license key data...
```

set Arguments:

[-i *key*]	Install a new license key.
[-r]	Remove license keys of the type specified by -t. Cannot be used with -i.
[-t *maint*]	Remove all of a certain type of license key. Cannot be used with -i. The maint option specifies a license key for an upgrade.
[-u]	Update license key.

Note: Either -i or -r must be specified.

set Examples:

To install new license keys:

set system licensekey -i << *EOF*
<key data>
EOF

To remove license keys for an upgrade:

set system licensekey -r -t maint

To update a license key:

set system licensekey -u
success
ok,00

local-dir-access

When used with the **get** verb, the **local-dir-access** target shows whether user access to the local directory is *enabled* or *disabled*. When used with the **set** verb, this target controls whether user access to the local directory is *enabled* or *disabled*. When set to *disabled*, users cannot access the local directory, save **REDIAL** list and corporate directory entries to the local directory, or select entries from the local directory when creating meeting entries in the meetings directory.

get Arguments:

None

system

get Examples:

    get system local-dir-access
    enabled

    ok,00
    get system local-dir-access -V
    Local Dir Access
    enabled

    ok

set Arguments:

`<{enabled|disabled}>`	Select `disabled` to restrict user access to the local directory, including saving REDIAL list and corporate directory entries to the local directory or accessing the Skype contact list.

set Examples:

    set system local-dir-access *disabled*

    ok,00

mcu

When used with the **get** verb, the **mcu** target retrieves the multiway calling status of a system. When used with **set** verb, this target controls whether multiway calls can be placed and received. When multiway calling is disabled, the system can support only one voice or one video call.

get Arguments:

    None

get Examples:

    get system mcu
    enabled

    ok,00

    get system mcu -V
    Multiway Calls
    enabled

    ok

set Arguments:

<{enabled|disabled}>	Specify disabled to limit the number of connected callers to either one voice or one video call.

set Examples:

set system mcu *enabled*

ok,00

meetings-dir-access

When used with the **get** verb, the **meetings-dir-access** target shows whether user access to the meetings directory is enabled or disabled. When used with the **set** verb, this target controls whether user access to the meetings directory is enabled or disabled. When set to *disabled*, users cannot access the meetings directory.

get Arguments:

None

get Examples:

get system meetings-dir-access
enabled

ok,00

get system meetings-dir-access -V
Meetings Dir Access
enabled

ok

set Arguments:

<{enabled|disabled}>	Select disabled to restrict user access to the meetings directory.

set Examples:

set system meetings-dir-access *disabled*

ok,00

message

The `message` target specifies a pop-up dialog box with a message and button layout to appear in the user interface. The dialog box can be used to inform the users of impending system maintenance or other important news. This target applies to the `set` verb.

Arguments:

`[-b {yes|no| cancel|ok}]`	Specify the buttons to be present in the popup. A maximum of 3 buttons may be displayed, but each button can only appear once.
`[-e]`	Specify that the dialog box use the error icon (red triangle with exclamation point). The default button layout for this dialog is the OK button.
`[-i]`	Specify that the dialog box use the information icon (message page). This is the default dialog type and includes an OK button.
`[-q]`	Specify that the dialog use the question icon (a question mark). The default button layout includes the Yes and No buttons.
`[-t seconds]`	Specify the timeout interval (in number of seconds) for the dialog. The default timeout is 30 seconds.
`[-w]`	Specify that the dialog use the warning icon (yellow triangle with exclamation point). The default button layout for this dialog is the OK button.
`message`	Specify the message to place in the dialog. If including spaces, enclose the entire message in double quotes. To wrap the message at a specific point, insert `\n` at the desired location in the message.

Examples:

Shows an information dialog with the desired text:

set system message "*Hello World*"

`ok,00`

Shows an error dialog with a 45-second timeout period, ok, and cancel buttons:

```
set system message -b ok -b cancel -t 45 -e "Too Hot"
```

```
ok,00
```

The user response is available through the **get system message-status** command.

message-status

The **message-status** target retrieves the user response from the most recent pop-up message displayed. Results may include the following:

- yes (user pressed the dialog box's yes or ok button)
- no (user pressed the dialog box's no button)
- cancel (user pressed the dialog box's cancel or the remote's back button)
- timeout (dialog box timed out before the user responded)
- empty string (user has not yet responded and the dialog box has not yet timed out)

This target applies to the **get** verb.

Arguments:

None

Examples:

```
get system message-status
```

```
ok,00
```

```
get system message-status -V
Result
cancel
```

```
ok
```

model

The **model** target shows the OEM and model name for the platform. This target applies to the **get** verb.

Arguments:

None

system

Examples:

get system model
LifeSize, Passport

ok,00

get system model -V
OEM   Model

LifeSize   Passport

ok

name

When used with the **get** verb, the **name** target shows the current name for the device. This is the same value that appears in the user interface and on a connected phone. When used with the **set** verb, this target sets the device name.

get Arguments:

None

get Examples:

get system name
Conference Room

ok,00

get system name -V
System Name
Conference Room

ok

set Arguments:

<value>	Specify the name, enclosed in quotes, for the system.

set Examples:

set system name "Marketing Region 1"

ok,00

network-presentation

When used with the **get** verb, the **network-presentation** target shows whether LifeSize Passport can accept a network presentation. When used with the **set** verb, this target controls whether network presentation is enabled for LifeSize Passport.

get Arguments:

   None

get Examples:

   get system network-presentation
   on

   ok,00

   get system network-presentation -V
   Network Presentations
   off

   ok

set Arguments:

`<{on|off}>`	Specify `on` to enable network presentations.

set Examples:

   set system network-presentation *on*

   ok,00

number

When used with the **get** verb, the **number** target retrieves the voice telephone number associated with the device. This appears in the user interface. When used with the **set** verb, this target sets the voice telephone number associated with the device.

get Arguments:

   None

get Examples:

   get system number
   555-1212

   ok,00

system

get system number -V
System Phone Number
555-1212

ok

set Arguments:

<value>	Specify the voice telephone number for the system.

set Examples:

set system number *555-1212*

ok,00

out-of-box

When used with the **get** verb, the **out-of-box** target shows the current state of the initial configuration process that starts when a system is installed or reset to its default configuration settings. When used with the **set** verb, this target runs the initial configuration process or cancels an already running initial configuration process.

get Arguments:

None

get Examples:

get system out-of-box
enabled

ok,00

get system out-of-box -V
Out Of Box Setup
complete

ok

set Arguments:

`<{enabled|complete}>`	Specify `enabled` to rerun the initial configuration process or `complete` to disable an already running initial configuration process.

set Examples:

```
set system out-of-box enabled
```

```
ok,00
```

screen-saver

When used with the **get** verb, the **screen-saver** target retrieves the current configuration of the screen saver feature. When used with the **set** verb, this target changes the screen saver.

get Arguments:

   None

get Examples:

```
get system screen-saver
window
```

```
ok,00
```

```
get system screen-saver -V
Screen Saver
window
```

```
ok
```

set Arguments:

<{logo|window|none}>	Set the screen saver type. The logo argument shows a roving logo; window shows a roving window; and none specifies no screen saver. The logo, window, and none arguments are available on all models.

set Examples:

```
set system screen-saver logo
```

```
ok,00
```

serial-number

The **serial-number** target retrieves the serial numbers of the CPU board and system board within the codec. This target applies to the **get** verb.

Arguments

None

Examples:

```
get system serial-number
BKxxxxxxxxxx,BKxxxxxxxxxx


ok,00


get system serial-number -V
CPU Board       System Board

BKxxxxxxxxxx    BKxxxxxxxxxx


ok
```

telepresence

When used with the **get** verb, the **telepresence** target shows whether or not the system is in telepresence mode. In this mode the user interface does not appear in the display. When used with the **set** verb, this target enables or disables the system telepresence feature.

get Arguments:

None

get Examples:

```
get system telepresence
on


ok,00


get system telepresence -V
Telepresence
off


ok
```

set Arguments:

`<{on|off}>`	Turn the telepresence feature on or off

set Examples:

```
set system telepresence on

ok,00
```

thermal-state

The `thermal-state` target returns the current thermal status within the system. This target applies to the `get` verb. The possible states include the following:

- `normal` indicates that the codec is operating in the normal temperature range.

- `warning` indicates that the codec is operating above the normal temperature range. The codec fan speed adjusts automatically in an attempt to cool itself.

- `overheated` indicates that the codec is overheated and approaching the maximum temperature before the codec reboots in an attempt to cool itself.

- `shutdown` indicates that the codec has reached the maximum temperature and will reboot in a matter of seconds in an attempt to cool itself.

Warning: Temperatures that require the codec to reboot can permanently damage codec components. Ensure the room that houses the codec is properly ventilated and temperature controlled.

Arguments:

    None

Examples:

```
get system thermal-state
normal

ok,00

get system temperatures -V
Thermal State
warning

ok
```

uptime

The `uptime` target returns the amount of time that the system has been up in days, hours, minutes, and seconds. This target applies to the `get` verb.

Arguments:

None

Examples:

```
get system uptime
5,21,13,20

ok,00

get system uptime -V
Days      Hours      Minutes      Seconds

5         21         13           40

ok
```

version

The `version` target returns the software version for all of the software loaded on the system. This target applies to the `get` verb.

Arguments:

None

Examples:

```
get system version
Software Version,LS_PP1_4.3.0 (0)
U-Boot Version,U-Boot 1.3.4 LifeSize Passport v0.02
Camera 0 Base,0x80406
Camera 0 Head,0x80406

ok,00
```

```
get system version -V
Software Version        Value
Software Version        LS_PP1_4.3.0 (0)
U-Boot Version          U-Boot 1.3.4. LifeSize Passport, v0.02a
Camera 0 Base           0x80406
Camera 0 Head           0x80406

ok
```

video-number

When used with the **get** verb, the **video-number** target retrieves the video telephone number associated with the system. This number appears in the user interface. When used with the **set** verb, this target sets the video telephone number associated with the system.

get Arguments:

None

get Examples:

```
get system video-number
555-1213

ok,00


get system video-number -V
System Video Number
555-1213

ok
```

set Arguments:

`<value>`	Specify the new video telephone number for the system.

set Examples:

```
set system video-number 555-1213

ok,00
```

telnet

When used with the **get** verb, the **telnet** target retrieves the current state of telnet protocol support. LifeSize recommends that you disable **telnet** because it is an insecure protocol. If you must use telnet, place the system behind a firewall or other external security device. By default, the telnet protocol service is disabled.

When used with the **set** verb, this object enables or disables the telnet service in real time. Active telnet sessions are disconnected if the service is stopped without closing the sessions first.

get Arguments:

   None

get Examples:

   get telnet
   on

   ok,00

   get telnet -V
   Telnet Service
   off

   ok

set Arguments:

`<{off|on}>`	Disable or enable the telnet service.

set Examples:

   set telnet *on*

   ok,00

timer

The following targets apply to the **timer** object.

caller-id

When used with the **get** verb, the **caller-id** target retrieves the current setting of the caller ID display timeout (the time in seconds before the caller ID display fades out). When used with the **set** verb, this target controls the time in seconds before the caller ID display fades out.

get Arguments:

   None

get Examples:

```
get timer caller-id
30

ok,00

get timer caller-id -V
Timeout in Seconds
on

ok
```

set Arguments:

<{off|5|15|30|60|120|300|600|on}>	Specify the time in seconds before the caller ID display fades out. Specifying off disables the caller ID display. Specifying on leaves the display on continuously.

set Examples:

```
set timer caller-id 30

ok,00
```

fadeout

When used with the **get** verb, the **fadeout** target retrieves the current setting for the user interface fadeout timer (the time in seconds before the user interface fades out during an active call). When used with the **set** verb, this target controls the time in seconds before the user interface fades out during an active call.

get Arguments:

   None

get Examples:

   get timer fadeout
   6

   ok,00

   get timer fadeout -V
   Timeout in Seconds
   5

   ok

set Arguments:

`<{5|10|20|30|60|120|300|600|never}>`	Specify the time in seconds before the user interface fades out during an active call. Specifying `never` disables the fadeout function.

set Examples:

   set timer fadeout *20*

   ok,00

   set timer fadeout *never*

   ok,00

screen-saver

When used with the **get** verb, the **screen-saver** target retrieves the current setting for the screen saver timer (the amount of idle time before the screen saver activates). When used with the **set** verb, this target controls the amount of idle time before the screen saver activates.

get Arguments:

  None

get Examples:

```
get timer screen-saver
20

ok,00

get timer screen-saver -V
Timeout in Minutes
10

ok
```

set Arguments:

`<{1|10|20|30|off}>`	Specify the time in minutes before the screen saver feature activates. Specifying `off` disables the screen saver function.

set Examples:

```
set timer screen-saver 20

ok,00

set timer screen-saver off

ok,00
```

timer

sleep

When used with the **get** verb, the **sleep** target retrieves the current setting for the sleep timer (the amount of idle time after the screen saver activates and before the system enters sleep state). When used with the **set** verb, this target sets the sleep timer.

get Arguments:

    None

get Examples:

```
get timer sleep
30

ok,00

get timer sleep -V
Timeout in Minutes
10

ok
```

set Arguments:

<{off|1|10|20|30}>	Specify the time in minutes before the system enters sleep state.
	WARNING: Specifying off disables system sleep. Damage may occur to the focus motor of LifeSize Camera causing the camera to fail if the system is awake for several hours in a dark room.

set Examples:

```
set timer sleep 20

ok,00
```

user

The **user** object enables configuration of user functions in the user interface.

password

The **password** target enables you to set the password for access to the user preferences in the user interface. This target applies to the **set** verb.

Arguments:

<value>	The new user password. The password must contain only the numbers 0-9 and/or the symbols * and #. The length can be 0 to 16 characters. If more than 16 characters are specified, the password is silently truncated.

Examples:

```
set user password 12345*#

ok,00

set user password -V abcdef

error  04  Invalid Parameter
```

verbose-mode

When used with the **get** verb, the **verbose-mode** target retrieves the current setting for verbose mode. When used with the **set** verb, this target enables or disables verbose mode output. Verbose mode provides human readable output. Enabling verbose mode is equivalent to specifying -V with each command entered.

get Arguments:

None

get Examples:

```
get verbose-mode
on

ok,00
```

video

```
get verbose-mode -V
Mode
off

ok
```

set Arguments:

`<{on|off}>`	Enable or disable verbose mode output.

set Examples:

```
set verbose-mode on

ok

set verbose-mode off

ok,00
```

video

The following targets apply to the **video** object.

background

When used with the **get** verb, the **background** target lists the available background image names for use with the **set video primary-background** commands. The list that appears is not sorted.

When used with the **set** verb, this target uploads a background image to the system. Before uploading a background image, ensure that the image has the following properties:

- 1280x720 pixels
- JPEG format
- base64 encoded

This command is intended for use only in conjunction with the system restore feature when the system state was exported through the **get config** command or through the web administration interface. As such, no validation is performed on the input data stream and no limitation is made as to the size of the image file. You many need to reset your system to defaults if you upload an invalid image file.

get Arguments:

　　None

get Examples:

```
get video background
1,European Subway
2,European Town
3,Lighthouse
4,Road

ok,00
```

```
get video background -V
Number          Background Image
1               European Subway
2               European Town
3               Lighthouse
4               Road

ok
```

set Arguments:

<image>	The name of the image file to save. You cannot replace the standard image files or the default image file.

set Examples:

```
unix% base64 --wrap=0 image | ssh auto@ip set video background
    image

ok,00
```

This command supports here document input or simple redirection:

```
set video background image << EOF
<base 64 encoded data stream>
EOF

ok,00
```

video

encode-quality

When used with the **get** verb, the **encode-quality** target retrieves the video encoder quality setting. When used with the **set** verb, this target controls the encoder quality setting. At higher settings, the encoder decreases transmitted resolution to increase video quality.

get Arguments:

> None

get Examples:

> **get video encode-quality**
> *-2*
>
> ok,00
>
> **get video encode-quality -V**
> Encoder Quality
> *0*
>
> ok

set Arguments:

`<{-4..4}>`	Specify the encoder quality setting. The default is `-2`. To specify a negative number, `--` must precede the value.

set Examples:

> **set video encode-quality -- *-4***
>
> ok,00
>
> **set video encode-quality *4***
>
> ok,00

input-names

When used with the **get** verb, the **input-names** target retrieves the display name associated with the video input. When used with the **set** verb, this target specifies the user friendly name of the video input.

get Arguments:

> None

get Examples:

```
get video input-names
HD Camera 1

ok,00


get video input-names -V
hd0
HD Camera 1

ok
```

set Arguments:

`<{hd0}>`	Specify the input to name. `hd0`, the only option, is for naming HD Camera 1.
`<value>`	The new name of the input. Enclose strings with spaces inside single or double quote characters.

set Examples:

```
set video input-names hd0 "HD Camera"

ok,00
```

layout

When used with the **get** verb, the **layout** target retrieves the value associated with the video layout in the active call. When used with the **set** verb, this target specifies a video layout to use in the active call.

Note: This target is intended for use only during an active call.

get Arguments:

None

get Examples:

```
get video layout
3

ok,00
```

video

```
get video layout -V
Current Video Layout
6

ok
```

set Arguments:

`<1..20>`	Specify the layout to use. Values not in this range return an error. Values greater than the maximum number of layouts available in the call within this range are ignored. To discover the maximum number of layouts available in the call, use the `get video max-layout` command.

set Examples:

```
set video layout 3

ok,00
```

layout-type

The `layout-type` target retrieves the screen layout type of the most recent or current call. The screen layout type determines the number of screen layouts that are available during a call. The following table describes the layout types and corresponding parameters.

Layout Type	Description
S1_HD	call video with 1 display
S2R_HD	call video and receiving a presentation with 1 display

This target applies to the `get` verb.

Arguments:

    None

Examples:

```
get video layout-type
S1_HD

ok,00
```

```
get video layout-type -V
Current type of video layout
S1_HD

ok
```

max-layout

The `max-layout` target retrieves the maximum number of video layouts supported by the active call. This target applies to the `get` verb.

Note: This target is intended for use during an active call. If the system is not in a call, the maximum number returned is 0.

Arguments:

   None

Examples:

```
get video max-layout
3

ok,00


get video max-layout -V
Current number of video layouts
6
ok
```

mtu

When used with the `get` verb, the `mtu` target retrieves the current setting of the video maximum transfer unit (MTU). When used with the `set` verb, this target sets the maximum transfer unit size in bytes for the video encoder.

get Arguments:

   None

get Examples:

```
get video mtu
1200

ok,00
```

video

```
get video mtu -V
Video MTU Size (bytes)
1440

ok
```

set Arguments:

<{900..1500}>	Specify the MTU for the video encoder.

set Examples:

```
set video mtu 1500

ok,00
```

pip-mode

When used with the **get** verb, the **pip-mode** target retrieves the current state of the picture-in-picture (PIP) feature. When used with the **set** verb, this target controls the default operation of the PIP feature during an active call. In auto mode, the window is visible only when the interface is visible and follows the fadeout timer settings. When on, the window is always visible; when off, it is never visible. This command cannot be used to change the PIP window state for an active call. To change the PIP window state for an active call, use the **pip-window** target.

get Arguments:

> None

get Examples:

```
get video pip-mode
auto

ok,00

get video pip-mode -V
Mode
on

ok
```

set Arguments:

<{on|off|auto}>	Specify how the PIP window operates.

set Examples:

> **set video pip-mode *auto***
>
> ok,00
>
> **set video pip-mode *off***
>
> ok,00

pip-window

The **pip-window** target controls the PIP display during a call. Turning the window on or off also changes the **pip-mode** setting to match the **pip-window** setting. This command shows the interface on the screen. If you set the PIP window to off, the window disappears when the interface fades out. If the interface fadeout timer is set to never, you cannot turn off the PIP window (because the interface never fades out). Turning the window on takes effect immediately. This target applies to the **set** verb.

Arguments:

`<{on|off}>`	Turn the PIP window on or off.

Examples:

> **set video pip-window *on***
>
> ok,00
>
> **set video pip-window *off***
>
> ok,00

primary-background

When used with the **get** verb, the **primary-background** target shows the name of the background image that appears in the primary display or **none** if a background color is used instead of an image. When used with the **set** verb, this target changes the background image or specifies that no background image appear in the primary display.

get Arguments:

> None

video

get Examples:

> **get video primary-background**
> *European Subway*
>
> ok,00
>
> **get video primary-background -V**
> Background Image
> *Road*
>
> ok

set Arguments:

<{*image*|none}>	Specify the background image to show or specify none to set a background color. The image may be the index number from **get video background**, any unique part of the image name from the same command (case insensitive), or default to show the default image. Specify none to choose a background color instead of an image. The default background color (sky blue) appears in the display. Use the **set video primary-background-color** command to specify a different background color.

set Examples:

> **set video primary-background** *town*
>
> ok,00
>
> **set video primary-background** *3*
>
> ok,00
>
> **set video primary-background** *none*
>
> ok,00

　　　　　　　LifeSize Passport Automation Command Line Interface

primary-background-color

When used with the **get** verb, the **primary-background-color** target shows the background color that appears in the primary display when the value of **get video primary-background** is *none*. When used with the **set** verb, this target specifies the background color for the primary display.

get Arguments:

   None

get Examples:

```
get video primary-background-color
sky-blue
ok,00
```

```
get video primary-background-color -V
Background Color
sky-blue
```

```
ok
```

set Arguments:

< {sky-blue|red|orange|yellow|brown|olive-green|green|dark-green|indigo|violet|light-gray|medium-gray|dark-gray|black}>	Specify the background color for the primary display when **get video primary-background** is *none*. The default color is sky-blue.
[-H #FFFFFF]	Specify a custom background color using a hexadecimal triplet. First character must be #, followed by 2 hexadecimal digits for the red value, then 2 hexadecimal digits for the green value, and finally 2 hexadecimal digits for the blue value. Cannot be used with -I.
[-I "{0..255} {0..255} {0..255}"]	Specify a custom background color using three integer values. The quotes are required. The first integer represents the red value, second integer represents the green value, and last integer represents the blue value. Cannot be used with -H.

video

`[-N "`*`color_name`*`"]`	Specify a name to display for your custom color. For use with `-I` and `-H`.

set Examples:

Set the primary background image to none:

**`set video primary-background `*`none`*

`ok,00`

Change the primary background color to light gray:

**`set video primary-background-color `*`light-gray`*

`ok,00`

Change the primary background color to a custom color:

`set video primary-background-color -I "`*`13 57 242`*`"`

`ok,00`

Change the primary background color to a custom color and name it:

`set video primary-background-color -H #`*`123456`*` -N "true-blue"`

`ok,00`

primary-display

When used with the **get** verb, the **primary-display** target shows the configuration for the primary display. When used with the **set** verb, this target controls the configuration of the primary video display.

get Arguments:

None

get Examples:

```
get video primary-display
auto,off

ok,00

get video primary-display -V
Resolution     Energy Saver
auto           off

ok
```

set Arguments:

`[-e {on|off}]`	Enables or disables the display energy saver feature. Specify **on** to enable this feature, which turns off the signal that the LifeSize system sends to the display when the system goes to sleep. LifeSize recommends that you test this feature for compatibility with your displays before using it in your environment. For more information, refer to the *LifeSize Passport User Guide*.
`[-r {auto|720 |768}]`	Specify the resolution of the primary display. Choose `720` for 1280x720 and `768` for 1280x768.

set Examples:

```
set video primary-display -r 768

ok,00
```

video

primary-display-override

When used with the **get** verb, this target retrieves the display type override specified by the **set primary-display-override** command. The default is *none*, which indicates no override is specified. Specify a display type override with the **set** verb as a troubleshooting measure when the primary display shows no video or distorted video, and the suspected cause may be an issue with the display cable, a cable adapter, or the configuration information provided by the display.

get Arguments:

   None

get Examples:

   No override specified (the default):

   get video primary-display-override
   none

   ok,00

   get video primary-display-override -V
   Override Display Type
   none

   ok

set Arguments:

<{none|hdmi| dvi|}>	Specify the display type override for the primary display. The default, none, specifies no override—the Lifesize system relies on the information sent from the display or uses the dvi override if no information is sent from the display. Specify dvi for 720/768p DVI.

set Examples:

   set video primary-display-override *dvi*

   ok,00

primary-input

When used with the **get** verb, the **primary-input** target retrieves the current setting for the primary input. When used with the **set** verb, this target controls what video source is associated with the primary input.

get Arguments:

> None

get Examples:

```
get video primary-input
hd0
ok,00

get video primary-input -V
Input
hd0

ok
```

set Arguments:

`<{hd0}>`	Specify the new source for the primary video input. The only valid value is `hd0`.

set Examples:

```
set video primary-input hd0

ok,00
```

video

primary-motion

When used with the **get** verb, the **primary motion** target retrieves the current setting for the preference of motion over sharpness when encoding the primary video stream. When used with the **set** verb, this target controls the preference for motion over sharpness when encoding the primary video stream.

get Arguments:

> None

get Examples:

> **get video primary-motion**
> *10*
>
> ok,00
>
> **get video primary-motion -V**
> Primary Video Motion
> *9*
>
> ok

set Arguments:

`<{1..10}>`	Specify the motion preference. Larger numbers prefer motion over sharpness.

set Examples:

> **set video primary-motion** *9*
>
> ok,00

streaming

When used with the **get** verb, the **streaming** target shows the current settings for streaming and recording. When used with the **set** verb, the target modifies the streaming settings.

get Arguments:

   None

get Examples:

   get video streaming
   disabled,,443,

   ok,00

   get video streaming -V
   State   Streaming IP Address   Port   Recording Key
   disabled   443

   ok

set Arguments:

`<{enabled|disabled}>`	Specify whether to enable or disable recording on this video communications system.
`<ip>`	Specify the IP address or hostname for the streaming server.
`[{1..65535}]`	Specify the TCP port number for the streaming server. The default, 443, is sufficient unless your network uses NAT with port forwarding rules that remap port 443 between the video communications system and LifeSize Video Center. This argument must reflect the remapped port number.
`[{0..99999}]`	Specify a default recording key to use for the recording with this video communications system.

set Examples:

   set video streaming *enabled videocenter.example.com 443 6513*

   ok,00

volume

The following targets apply to the **volume** object.

dtmf

When used with the **get** verb, the **dtmf** target retrieves the current volume setting (using a scale of 0 to 10) for Dual Tone Multi Frequency (DTMF) tones when placing a call. When used with the **set** verb, this target controls the volume setting for DTMF tones.

get Arguments:

> None

get Examples:

```
get volume dtmf
5

ok,00

get volume dtmf -V
Volume
6

ok
```

set Arguments:

`<{0..10}>`	Specify the volume level (0 = off, 10 = max) for DTMF tone generation.

set Examples:

```
set volume dtmf 5

ok,00
```

ring-tone

When used with the **get** verb, the **ring-tone** target retrieves the current volume setting for the ring tone. When used with the **set** verb, this target controls the volume setting for the ring tone.

get Arguments:

    None

get Examples:

```
get volume ring-tone
5

ok,00

get volume ring-tone -V
Volume
6

ok
```

set Arguments:

`<{0..10}>`	Specify the volume level (0 = off, 10 = max) for ring tone generation.

set Examples:

```
set volume ring-tone 5

ok,00
```

speaker

When used with the **get** verb, the **speaker** target retrieves the current volume setting for the system speaker (audio loudness). When used with the **set** verb, this target controls the volume of the system speaker.

get Arguments:

    None

get Examples:

```
get volume speaker
50

ok,00
```

volume

```
get volume speaker -V
Volume
70

ok
```

set Arguments:

`<{0..100}>`	Specify the volume level (0 = off, 100 = max) for system audio.

set Examples:

```
set volume speaker 60
ok,00
```

status-tone

When used with the `get` verb, the `status-tone` target retrieves the current volume setting for the system status tones. When used with the `set` verb, this target controls the volume of the system status tones.

get Arguments:

    None

get Examples:

```
get volume status-tone
3

ok,00

get volume status-tone -V
Volume
5

ok
```

set Arguments:

`<{0..10}>`	Specify the volume level (0 = off, 10 = max) for the status tones.

set Examples:

```
set volume status-tone 5

ok,00
```

status Verb: Object and Targets

The following objects and targets apply to the **status** verb.

call

The following targets apply to the **status call** object.

active

The **active** target shows the status of all active calls in the system.

Arguments:

`[-C call]`	Restrict output to the specified call ID.
`[-d incoming|outgoing]`	Restrict output to the specified call direction.
`[-t audio|video]`	Restrict output to the specified call type.

Examples:

```
status call active
1,1,Connected,No,Video,10.10.11.166,Techpubs,Yes,h323,No,0:02:23

ok,00
```

Note: The output in the following example is split by column into tables for visual clarity. The actual output is a single line for each call.

```
status call active -V -d incoming -t video
```

Call	Conf	State	Incoming	Type	Number
5	*1*	*Connected*	*Yes*	*Video*	*10.10.11.166*

Name	Muted	Protocol	Secure	Duration
Techpubs	*Yes*	*h323*	*No*	*0:02:29*

```
ok
```

Note: Valid values for the `State` field are *Dialing*, *Ringing*, *Connected*, *Terminating*, and *Ringback*. Valid values for the `Type` field are *Video*, *Audio*, and *Unknown*.

call

history

The **history** target shows historical information on completed calls. Active calls are not shown. The information consists of the following fields:

Field Number	Field Name	Description	Display Mode[a]
1	ID	The call identifier—a monotonically incrementing index for the call **Note**: This is not the call handle used during an active call.	Default
2	Conf	The conference identifier— a monotonically incrementing index used to distinguish conference participants **Note**: This is not the conference handle used during an active conference.	Default
3	Local Name	The name of the local system (the system providing the call history)	Default
4	Local Number	The number of the local system	Default
5	Remote Name	The name of the remote system (the other participant in the call)	Default
6	Remote Number	The number of the remote system	Default
7	Dialed Digits	The digits used to place the call	Default
8	Start Time	The time in ISO date format at which the call started	Default
9	End Time	The time in ISO date format at which the call ended	Full
10	Duration	The length of the call in hours:minutes:seconds	Default
11	Direction	Indication of incoming or outgoing call	Default
12	Protocol	The communications protocol used for the call	Full
13	Security	The security protocol used for the call	Full
14	Req Kibps	Requested bit rate for the call	Full
15	Act Kibps	Actual bit rate for the call	Full
16	TX Vid	Transmit video codec used	Full

Field Number	Field Name	Description	Display Mode[a]
17	TX Aud	Transmit audio codec used	Full
18	TX Res	Transmit resolution used	Full
19	RX Vid	Received video codec used	Full
20	RX Aud	Received audio codec used	Full
21	RX Res	Received resolution	Full
22	TX Pres	Transmit presentation status—whether or not a presentation was transmitted	Full
23	RX Pres	Receive presentation status—whether or not a presentation was received	Full
24	Pres Fmt	Presentation format—the protocol used for the secondary video stream	Full
25	Term Code	Call termination code	Full
26	TxV1 Pct Loss	Percent packet loss of primary video transmitted	Full
27	RxV1 Pct Loss	Percent packet loss for primary video received	Full
28	TxV1 Pkts Lost	Number of packets lost for primary video transmitted	Full
29	RxV1 Pkts Lost	Number of packets lost for primary video transmitted	Full
30	TxV1 Avg Jitter	Average jitter for primary video transmitted	Full
31	RxV1 Avg Jitter	Average jitter for primary video received	Full
32	TxV1 Max Jitter	Maximum jitter for primary video transmitted	Full
33	RxV1 Max Jitter	Maximum jitter for primary video received	Full
34	TxA1 Pct Loss	Percent packet loss for audio transmitted	Full
35	RxA1 Pct Loss	Percent packet loss for audio received	Full
36	TxA1 Pkts Lost	Number of packets lost for audio transmitted	Full

call

Field Number	Field Name	Description	Display Mode[a]
37	RxA1 Pkts Lost	Number of packets lost for audio received	Full
38	TxA1 Avg Jitter	Average jitter for audio transmitted	Full
39	RxA1 Avg Jitter	Average jitter for audio received	Full
40	TxA1 Max Jitter	Maximum jitter for audio transmitted	Full
41	RxA1 Max Jitter	Maximum jitter for audio received	Full
42	TxV2 Pct Loss	Percent packet loss for secondary video transmitted	Full
43	RxV2 Pct Loss	Percent packet loss for secondary video received	Full
44	TxV2 Pkts Lost	Number of packets lost for secondary video transmitted	Full
45	RxV2 Pkts Lost	Number of packets lost for secondary video transmitted	Full
46	TxV2 Avg Jitter	Average jitter for secondary video transmitted	Full
47	RxV2 Avg Jitter	Average jitter for secondary video received	Full
48	TxV2 Max Jitter	Maximum jitter for secondary video transmitted	Full
49	RxV2 Max Jitter	Maximum jitter for secondary video received	Full

a. The `Default` display mode indicates that the field always appears in the output. The `Full` display mode indicates that the field appears in the output only when you specify the `-f` option with the command. By default, only a limited set of statistics appear.

Arguments:

-U	Show times as UTC instead of local time.
-f	Enable full display mode showing all available statistics.
-X	Shows data for up to 1000 calls. When this argument is not specified, the maximum number of calls for which data is shown is 26. This argument cannot be used with -V.

Examples:

status call history
39,10,lifesize,10.10.11.209,unknown,9710,1234567,2007-07-09 17:13:32,01:02:56,In
38,10,lifesize,10.10.11.209,unknown,1310,8901234,2007-07-09 16:14:03,00:00:00,In
37,9,lifesize,10.10.11.209,unknown,9710,5678901,2007-07-09 15:13:13,00:16:45,In

ok,00

Note: The output in the following example is split by column into tables for visual clarity. The actual output is a single line for each call.

status call history -V

ID	Conf	Local Name	Local Number	Remote Name	Remote Number	Dialed Digits
39	*10*	*lifesize*	*10.10.11.209*	*unknown*	*9710*	*1234567*
38	*10*	*lifesize*	*10.10.11.209*	*unknown*	*1310*	*8901234*
37	*9*	*lifesize*	*10.10.11.209*	*unknown*	*9710*	*5678901*

Start Time	Duration	Direction
2007-07-09 17:13:32	*01:02:56*	*In*
2007-07-09 16:14:03	*00:00:00*	*In*
2007-07-09 15:13:13	*00:16:45*	*In*

ok

status call history -f
39,10,lifesize,10.10.11.209,unknown,9710,1234567,
   2007-07-09 17:13:32,2007-07-09,18:16:28,01:02:56,In,H.323,
   None,512,448,H.264,G.711Ulaw,HD,H.264,G.711Ulaw,HD,No,No,
   None,Normal,0.000,0.000,0,0,5.000,5.000,16,9,0.000,0.000,0,0,
   31.000,31.000,31,31,0.000,0.000,0,0,0.000,0.000,0,0

38,10,lifesize,10.10.11.209,unknown,1310,8901234,
   2007-07-09 16:14:03,2007-07-09 16:14:03,00:00:00,In,H.323,
   None,1152,0,,,,,,,No,No,None,Normal,0.000,0.000,0,0,5.000,
   5.000,16,9,0.000,0.000,0,0,31.000,31.000,31,31,0.000,0.000,0,
   0,0.000,0.000,0,0

call

```
37,9,lifesize,10.10.11.209,unknown,9710,5678901,
  2007-07-09 15:13:13,2007-07-09 15:29:58,00:16:45,In,H.323,
  None,512,448,H.264,G.711Ulaw,HD,H.264,G.711Ulaw,HD,No,No,
  None,Normal,0.000,0.000,0,0,5.000,5.000,16,9,0.000,0.000,0,0,
  31.000,31.000,31,31,0.000,0.000,0,0,0.000,0.000,0,0
```

```
ok,00
```

Note: The output in the following example is split by column into tables for visual clarity. The actual output is a single line for each call.

status call history -V -f

ID	Conf	Local Name	Local Number	Remote Name	Remote Number	Dialed Digits
39	10	lifesize	10.10.11.209	unknown	9710	1234567
38	10	lifesize	10.10.11.209	unknown	1310	8901234
37	9	lifesize	10.10.11.209	unknown	9710	5678901

Start Time	End Time	Duration	Direction	Protocol	Security
2007-07-09 17:13:32	2007-07-09 18:16:28	01:02:56	In	H.323	None
2007-07-09 16:14:03	2007-07-09 16:14:03	00:00:00	In	H.323	None
2007-07-09 15:13:13	2007-07-09 15:29:58	00:16:45	In	H.323	None

Req Kibps	Act Kibps	TX Vid	TX Aud	TX Res	RX Vid	RX Aud	RX Res
512	448	H.264	G.711Ulaw	HD	H.264	G.711Ulaw	HD
1152	0						
512	448	H.264	G.711Ulaw	HD	H.264	G.711Ulaw	HD

TX Pres	RX Pres	Pres Fmt	Term Code	TxV1 Pct Loss	RxV1 Pct Loss	TxV1 Pkts Lost
No	No	None	Normal	0.000	0.000	0
No	No	None	Normal	0.000	0.000	0
No	No	None	Normal	0.000	0.000	0

RxV1 Pkts Lost	TxV1 Avg Jitter	RxV1 Avg Jitter	TxV1 Max Jitter	RxV1 Max Jitter
0	5.000	5.000	16	9
0	5.000	5.000	16	9
0	5.000	5.000	16	9

TxA1 Pct Loss	RxA1 Pct Loss	TxA1 Pkts Lost	RxA1 Pkts Lost	TxA1 Avg Jitter
0.000	0.000	0	0	31.000
0.000	0.000	0	0	31.000
0.000	0.000	0	0	31.000

```
RxA1 Avg Jitter TxA1 Max Jitter RxA1 Max Jitter TxV2 Pct Loss   RxV2 Pct Loss
31.000          31              31              0.000           0.000
31.000          31              31              0.000           0.000
31.000          31              31              0.000           0.000


TxV2 Pkts Lost  RxV2 Pkts Lost  TxV2 Avg Jitter RxV2 Avg Jitter TxV2 Max Jitter
0               0               0.000           0.000           0
0               0               0.000           0.000           0
0               0               0.000           0.000           0


RxV2 Max Jitter
0
0
0

ok
```

statistics

The **statistics** target shows bandwidth and codec statistics for active calls or a specific active call.

Following are the complete set of fields that appear.

Field Number	Field Name	Description
1	ID	The call handle.
2	ARX Codec	Audio Receive Codec – shows the audio codec used by the remote transmitter.
3	Kibps	Kilobits per second – shows the bit rate divided by 1024 for the preceding column's codec.
4	ATX Codec	Audio Transmit Codec – shows the audio codec used by the local transmitter.
5	Kibps	Kilo Bits per second – shows the bit rate divided by 1024 for the preceding column's codec.
6	VRX Codec	Video Receive Codec – shows the video codec used by the remote transmitter.

call

Field Number	Field Name	Description
7	Kibps	Kilobits per second – shows the bit rate divided by 1024 for the preceding column's codec.
8	VTX Codec	Video Transmit Codec – shows the video codec used by the local transmitter.
9	Kibps	Kilobits per second – shows the bit rate divided by 1024 for the preceding column's codec.
10	ARX Jitter	Audio Receive Jitter – shows the packet jitter from the remote audio transmission.
11	ARX Pktps	Audio Receive Packets per second – shows the received audio packet rate which is dependent on the bit rate and codec used.
12	ARX Pkt Loss	Audio Receive Packet loss – shows the instantaneous number of audio packets transmitted by the remote side that were never received (or received too late) at the local side.
13	ARX Cumu Loss	Audio Receive cumulative packet loss – shows the total number of remote transmitted audio packets that were lost.
14	ARX % Loss	Audio Receive percentage packet loss – shows the percent of the total remote transmitted audio packets that were lost.
15	ATX Jitter	Audio Transmit Jitter – shows the packet jitter from the local audio transmission.
16	ATX Pktps	Audio Transmit Packets per second – shows the transmitted audio packet rate which is dependent on the bit rate and codec used.
17	ATX Pkt Loss	Audio Transmit Packet loss – shows the instantaneous number of audio packets transmitted by the local side that were never received (or received too late) at the remote side.
18	ATX Cumu Loss	Audio Transmit cumulative packet loss – shows the total number of locally transmitted audio packets that were lost.
19	ATX % Loss	Audio Transmit percentage packet loss – shows the percent of the total locally transmitted audio packets that were lost.
20	VRX Jitter	Video Receive Jitter – shows the packet jitter from the remote video transmission.

Field Number	Field Name	Description
21	VRX Pkt Loss	Video Receive Packet loss – shows the instantaneous number of video packets transmitted by the remote side that were never received (or received too late) at the local side.
22	VRX Cumu Loss	Video Receive cumulative packet loss – shows the total number of remote transmitted video packets that were lost.
23	VRX % Loss	Video Receive percentage packet loss – shows the percent of the total remote transmitted video packets that were lost.
24	VRX Fps	Video Receive Frames per second – shows the frame rate of the received video.
25	VRX Res	Video Receive Resolution – shows the resolution (width by height) of the received video.
26	VTX Jitter	Video Transmit Jitter – shows the packet jitter from the local video transmission.
27	VTX Pkt Loss	Video Transmit Packet loss – shows the instantaneous number of video packets transmitted by the local side that were never received (or received too late) at the remote side.
28	VTX Cumu Loss	Video Transmit cumulative packet loss – shows the total number of local transmitted video packets that were lost.
29	VTX % Loss	Video Transmit percentage packet loss – shows the percent of the total number of local transmitted video packets that were lost.
30	VTX Fps	Video Transmit Frames per second – shows the frame rate of the transmitted video.
31	VTX Res	Video Transmit Resolution – shows the resolution (width by height) of the transmitted video.

Arguments:

[-C callHandle]	Specify that statistics for a specific call handle are desired. This argument cannot be used with **-a**.

call

Examples:

status call statistics

1,G722,78.1,AAC_LC,94.0,H264,924.6,H264,893.9,19,50,0,0,
    0.000000,19,50,0,0,0.000000,7,0,0,0.000000,30,
    1280 720,7,0,0,0.000000,30,1280 720
4,G711ULAW,62.4,G711ULAW,62.4,H264,1050.2,H264,1050.2,17,50,0,0,
    0.000000,17,50,0,4,0.000000,9,0,0,0.000000,30,
    1280 720,9,0,0,0.000000,30,1280 720

ok,00

Note: The output in the following example is split by column into tables for visual clarity. The actual output is a single line for each call.

status call statistics –V

ID	ARX Codec	Kibps	ATX Codec	Kibps	VRX Codec	Kibps	VTX Codec
1	*G722*	*78.1*	*AAC_LC*	*94.0*	*H264*	*924.6*	*H264*
4	*G711ULAW*	*62.4*	*G711ULAW*	*62.4*	*H264*	*1050.2*	*H264*

Kibps	ARX Jitter	ARX Pktps	ARX Pkt Loss	ARX Cumu Loss	ARX % Loss
893.9	*19*	*50*	*0*	*0*	*0.000000*
1050.4	*17*	*50*	*0*	*0*	*0.000000*

ATX Jitter	ATX Pktps	ATX Pkt Loss	ATX Cumu Loss	ATX % Loss	VRX Jitter
19	*50*	*0*	*4*	*0.000000*	*7*
17	*50*	*0*	*4*	*0.000000*	*9*

VRX Pkt Loss	VRX Cumu Loss	VRX % Loss	VRX Fps	VRX Res	VTX Jitter
0	*0*	*0.000000*	*30*	*1280 720*	*7*
0	*0*	*0.000000*	*30*	*1280 720*	*9*

VTX Pkt Loss	VTX Cumu Loss	VTX % Loss	VTX Fps	VTX Res
0	*0*	*0.000000*	*30*	*1280 720*
0	*0*	*0.000000*	*30*	*1280 720*

ok

```
status call statistics -C 1
1,G722,62.4,AAC_LC,93.7,H264,924.6,H264,894.8,19,50,0,0,
   0.000000,19,50,0,0,0.000000,7,0,0,0.000000,30,
   1280 720,7,0,0,0.000000,30,1280 720

ok,00
```

presentation

The **presentation** object shows information about presentations.

statistics

The **statistics** target shows information about active received presentations.

Arguments:

[-c conference]	Restrict output to the specified conference ID.

Examples:

```
status presentation statistics
1,true,rx,dec2,H264,145.6,1280 720

ok,00
```

```
status presentation statistics -V
Conf      Enabled   Type     Device    Codec    Kibps     Resolution
1         true      rx       sd0       H264     144.7     704 480

ok
```

control Verb: Objects and Targets

The following objects and targets apply to the **control** verb.

Asynchronous Messages

During normal operation, the system may receive asynchronous messages relating to call status changes, presentation status changes, or incoming call notifications. These messages are printed after a command completes between the ok or error message and the shell prompt, for example:

```
...
ok,00
CS,3,1,Connected,Video,Normal,10.10.11.10,LifeSize
<prompt>
```

In addition, if the shell detects that no input has been received since the prompt was printed, it may spontaneously print an asynchronous message by emulating the user having pressed return. In this way, asynchronous messages are delivered in a timely fashion while still guaranteeing that the messages do not interfere with processing the current command being executed.

Because asynchronous messages may be received at any time and to preserve the order in which messages arrive, some commands do not produce any synchronous output and instead produce only asynchronous output. Execution of these commands generally causes asynchronous messages (for example, placing a call, starting a presentation). Commands that operate in this fashion are indicated as doing so in the description of the command.

Call Status Messages

While a call is active, or as a response to the **control call**, **answer**, **dial**, and **hangup** commands, the CLI produces status messages about the call. These messages use the Call Status (CS) format. For example:

```
...
ok,00
CS,3,1,Connected,Video,Normal,10.10.11.10,LifeSize
<prompt>
```

The meaning of the columns is as follows:

Col #	Meaning	Values	Description
1	Prefix	CS	CS indicates that this asynchronous event is a call status update.
2	Call ID	<number>	Indicates the number of the call.
3	Conference ID	<number>	Indicates the number of the conference managing this call.
4	State	On Hook	Phone is on hook.
		Terminating	Call is terminating.
		Terminated	Call is terminated (but may still be off hook).
		Off Hook	Phone is off hook.
		Valid Number	Dialed number is valid.
		Dialing	Dialing is proceeding.
		Proceeding	Call is proceeding.
		Ringing	Call is ringing.
		Answered Number	Answered number information.
		Answered Consult	When a call is answered in consult mode (private from main call).
		Connected	Call is connected.
		Call Encrypted	Call is encrypted.
		Call Not Encrypted	Call is not encrypted.
		Notify Info	Notification of miscellaneous events.
		Ring Incoming	Incoming call received.
		Caller ID	Caller ID information.
		Local Ring Back Off	Local ringback is off.
		Remote Pres Begin	A remote presentation is beginning.
		Remote Pres End	A remote presentation is ending.
		Remote Pres Failed	The remote presentation has failed.
		Far End Mute	The far end has muted the microphone.
		Far End Unmute	The far end has unmuted the microphone.
		Far End Hold	The far end has placed the call on hold.
		Far End Resume	The far end has resumed the call.
5	Type	Audio	The message pertains to an audio call.
		Video	The message pertains to a video call.
		Unknown	The message pertains to either type of call.

Call Status Messages

Col #	Meaning	Values	Description
6	Disconnect Reason	Normal	Normal disconnection.
		Unknown	Unknown reason for disconnection.
		Busy	Remote end is busy.
		No Answer	Remote end did not answer.
		Bad Number	Invalid number dialed.
		Comm Failure	Communications failure.
		Unreachable	Remote end is unreachable.
		Rejected	Remote end rejected the call.
		Max Calls	Simultaneous call limit reached.
		Parse Error	Parse error in called address.
		Enc Not Sup	Encoder not supported.
		No Bandwidth	No bandwidth available for call.
		Unreachable GK	Gatekeeper is unreachable.
		GK Resources	Gatekeeper out of resources for call.
		GW Resources	Gateway out of resources for call.
		Invalid Addr	Invalid called address.
		Not Registered	Called address not registered.
		SIP 400	SIP Bad request.
		SIP 403	SIP Disallowed.
		SIP 404	SIP Remote party not in a domain.
		SIP 415	SIP Mismatched codec.
		SIP 416	SIP Unsupported address.
		SIP 480	SIP User temporarily unavailable.
		SIP 500	SIP Server error.
		SIP 502	SIP Bad gateway.
		SIP 513	SIP Server failed - request too large.
		SIP 603	SIP User declined call.
		SIP 606	SIP Service not acceptable.
		No Audio Resources	Insufficient audio resources.
		Admission Control	Call rejected due to heavy call load.
		SR Maint Mode	Server in maintenance mode.
		SR Storage	Server is full.
		SR Max Calls	Server is beyond call capacity.
		SR Pin Invalid	Streaming and recording key invalid.
		SR Admin Disconnect	Administrator shut down server services.
7	Number	<IP>	The phone number of the remote side of the call.
8	Name	<string>	The assigned name of the remote system.

Incoming Call Messages

When an incoming call is received, a status message about the call is printed. These messages use mostly the same format as the call status messages previously described, but are prefixed with "IC" (incoming call) instead of "CS" and do not contain the disconnect reason field (column 6 in the previous example), for example:

```
...
ok,00
IC,16,1,Ringback,Video,10.10.11.155,Sunshine
<prompt>
```

Once an incoming call notice has been generated, further notices about that specific call are relayed through call status ("CS") messages.

Presentation Status Messages

While a presentation is active, the CLI produces status messages about the presentation. These messages use the PS (presentation status) format, for example:

```
...
ok,00
PS,15,1,Terminated,No,Rejected
<prompt>
```

The output columns for this command are as follows:

Col #	Meaning	Values	Description
1	Prefix	PS	PS indicates that this asynchronous event is a presentation status event.
2	Presentation ID	*<number>*	Indicates the number of the presentation.
3	Conference ID	*<number>*	Indicates the number of the conference running the presentation.
4	State	Initiated Terminated Relinquished	The presentation has started. The presentation has ended. The local presentation has been superseded by a remote one.
5	Remote	Yes	The presentation message concerns a remote presentation.

Far Camera Control Messages

Col #	Meaning	Values	Description
6	Disconnect Reason	None	No disconnect has occurred.
		Normal	Normal disconnect occurred (phone on hook).
		Rejected	The presentation was rejected.
		Unknown	Unknown disconnect occurred.

Far Camera Control Messages

During an active call you may receive control messages for the local camera. These messages use the FC (far camera) format. For example:

```
...
ok,00
FC,1,Near,main,Move,Pan Left  # Begin moving camera left.
...
ok,00
FC,1,Near,main,Move,Continue  # Continue current camera motion.
...
ok,00
FC,1,Near,main,Stop,None   # Stop camera movement.
```

Following are the output columns for this format:

Col #	Meaning	Values	Description
1	Prefix	FC	FC indicates this asynchronous event is a far camera control message.
2	Call ID	*<number>*	The call ID associated with this message.
3	Where	Near	The local camera is the target of the operation.
		Far	The far camera is the target of the operation.
		None	The message does not pertain to a camera.
4	Target	main	Indicates that the primary HD camera is available.
5	Operation	Move	The camera should begin or continue motion.
		Stop	The camera should stop all motion immediately.
		None	The message is not a motion control message.

Col #	Meaning	Values	Description
6	Movement	Pan Left	Pan the camera to the left.
		Pan Right	Pan the camera to the right.
		Tilt Up	Tilt the camera up.
		Tilt Down	Tilt the camera down.
		Zoom In	Zoom the camera in (telephoto).
		Zoom Out	Zoom the camera out (widen).
		Focus In	Focus the camera in.
		Focus Out	Focus the camera out.
		None	Not a camera motion operation.

Mute Status Messages

During normal call operation, the state of the remote side mute function is available through the call status messages. The local mute status is available through the Mute Status message. These messages use the MS prefix and are in direct response to the user pressing the local mute button on either the phone or the remote. The mute status messages use the following format:

```
...
ok,00
MS,true
...
ok,00
MS,false
```

When the second column is *true*, the local side is muted. When the second column is *false*, the local side is not muted. When an outgoing call is placed, the local mute status is false. When an incoming call is placed, the state of the auto-mute (see **get call auto-mute**) controls the initial state. The current status is available through the **get audio mute** command.

Video Capabilities Messages

The remote side of a call may support sending more than one video source. The video capabilities message provides a means to determine which sources are supported and what capabilities they provide. This message generally appears after a call is connected, but may also appear mid call if the remote codec supports hot-plugging of video sources. Video capabilities messages use the following format:

```
...
ok,00
VC,12,2,main,PTZF,auxdoc,---
...
ok,00
VC,12,1,main,PTZF
```

The output columns are as follows:

Col #	Meaning	Values	Description
1	Prefix	VC	VC indicates that this asynchronous event is a video capabilities message.
2	Call ID	*<number>*	The call ID associated with this message.
3	Count	*<number>*	The number of video sources available for this call. Each video source indicated in this count has two additional columns.

Col #	Meaning	Values	Description
4, 6, ...	Source	main	Indicates that the primary HD camera is available.
		aux	Indicates that the secondary HD camera is available. On LifeSize Express, this is a camera connected to the HD input. On LifeSize Room 200, LifeSize Room 220, LifeSize Team 200, and LifeSize Team 220, this is a camera connected to HD input 1.
		doc	Indicates that the document camera is available. Applies to LifeSize Room, LifeSize Team, and LifeSize Team MP.
		auxdoc	Indicates that the VGA input (LifeSize Room, LifeSize Team, LifeSize Team MP, and LifeSize Express) or the DVI-I input (LifeSize Room 200, LifeSize Room 220, LifeSize Team 200, LifeSize Team 220, LifeSize Express 200, and LifeSize Express 220) is available.
		playback	Indicates that the DVD/VCR input is available. Applies to LifeSize Room, LifeSize Room 200, and LifeSize Room 220.
		6..15	Indicates other video sources that may be available on the device. On a LifeSize system, the value 6 is a camera on HD input 2 on LifeSize Room 200 or LifeSize Room 220.
5, 7, ...	Capabilities	`PTZF` or `- - - -` or a combination	Each character is a flag indicating a capability supported. `P` indicates support for panning, `T` indicates support for tilting, `Z` indicates support for zooming and `F` indicates support for focusing. A dash (-) indicates that the corresponding capability is not present.

Only the supported sources are reported. If a source is not listed, video is not available from that source.

System Sleep Messages

System sleep messages are generated when a system goes to sleep and when it wakes up.

```
...
ok,00
SS,true
...
ok,00
SS,false
```

When the second column is *true*, the system has gone to sleep. When the second column is *false*, the system has awakened.

call

The following targets apply to the **call** object.

For those targets that take a called address, the address may be specified as an IP address, a URI, or a directory specification string. Following are the directory specification strings:

Form	Description
redial:*<n>*	Dial the indicated entry from the redial list. Entry 1 is at the top of the list, entry 2 is the entry immediately following it. The ordering of the redial list changes as calls are placed and received. For automation use, do not use this form unless the intent is to redial the last call.
redial:*<string>*	Dial the indicated entry from the redial list. The string is used as a case insensitive prefix to match the name stored in the redial list (the name that shows in the user interface). For example, the prefix "sun" matches the names "sunrise" and "SUNSET", but not "summer" or "fun-in-the-sun". For automation use, the prefix should completely specify the intended entry and that entry should be locked in the redial list.
local:*<string>*	Dial the indicated entry from the local directory. The string is used as a case insensitive prefix to match against the name stored in the local directory. For automation use, the string should completely specify the desired entry.
corp:*<string>*	Dial using the indicated entry from the corporate (network) directory.
meeting:*<string>*	Dial using the indicated entry from the meetings directory. Can only be used with **dial** to initiate a new meeting, not add a participant with **add-part**.

add-part

The **add-part** target places a conference call if one does not exist or adds a new participant to an existing conference call. This command produces only asynchronous messages as a response to ensure proper ordering of displayed call status.

Arguments:

`[-b {auto|128|192|256|`
`320|384|512|640|768|`
`896|1024|1152|1280|1472|`
`1728|1920|2000}]`
	Specify the maximum bandwidth in kilobits per second (kb/s) to use for the call. Use `auto`, the default, to use the configured maximum bandwidth.

`[-p {auto|h323|h323gw|` `isdn|sip|ip|skype|rtsp}]`	Specify the protocol to use to connect the new party to the call. The default is `auto`.
`[-t {audio|video}]`	Specify to add the participant as an audio or video call. The default is `video`. Specifying `video` fails if a video call is in progress.

<confHandle>	Specify the handle to the conference to which to add the participant. Specify 0 to place a conference call if one does not exist or to specify the active conference.
<number>	Specify the phone number, IP address, or URI of the party to add.

Examples:

set prompt "% "

ok,00

Initiate a call:

% control call add-part *0* -t *audio 10.10.11.166*

ok,00
CS,4,1,Dialing,Audio,Normal,10.10.11.166,10.10.11.166

Add a participant to an existing call:

% control call add-part -V *2* -p *h323 10.10.11.11* -b *1024*

ok
CS,*5,2,Ringing,Video,Normal,10.10.11.10,LifeSize*

Add 3rd redial entry as call:

% control call add-part *1 redial:3*

ok,00
CS,*3,1,Ringing,Audio,Normal,1-512-555-1212,*
%

Refer to "Asynchronous Messages" on page 212 for a description of the response.

answer

The **answer** target answers or rejects an incoming call. This command produces only asynchronous messages as a response to ensure proper ordering of displayed call status.

Arguments:

`[-r]`	Specify whether to reject the call. The default is to answer. Cannot be used with `-t`.
`[-t {audio|video}]`	Specify whether to answer the call as an audio or video call. Cannot be used with `-r`.
`<callHandle>`	Specify the handle of the incoming call to answer.

Examples:

```
set prompt "% "

ok,00
% control call answer 1 -t audio

ok,00
CS,1,2,Connected,Audio,Normal,10.10.11.10,LifeSize
% control call answer 2 -r -V

ok
CS,1,2,Terminated,Video,Rejected,10.10.11.10,LifeSize
%
```

Refer to "Asynchronous Messages" on page 212 for a description of the response.

del-part

The **del-part** target drops a participant from an existing conference call. This command produces only asynchronous messages as a response to ensure proper ordering of displayed call status.

Note: You cannot use this object to delete a participant when the call to that participant failed (for example, called number was busy). You must use control call hangup to delete that participant.

Arguments:

<callHandle>	Specify the handle to the call on which to drop the participant.

Examples:

```
set prompt "% "

ok,00
% control call del-part 3

ok,00
CS,3,2,Terminated,Video,Normal,10.10.11.10,LifeSize
% control call del-part -V 2

ok
CS,2,2,Terminated,Video,Normal,10.10.11.10,LifeSize
%
```

Refer to "Asynchronous Messages" on page 212 for a description of the response.

dial

The **dial** target initiates a new call. This command produces only asynchronous messages as a response to ensure proper ordering of displayed call status.

Arguments:

[-b {auto|128|192|256|320|384|512|640|768|896|1024|1152|1280|1472|1728|1920|2000}]	Specify the maximum bandwidth to use for the call. Use auto to use the configured maximum bandwidth. The default is auto.
[-p {auto|h323|h323gw|isdn|sip|ip|skype|rtsp}]	Specify the protocol to use to connect the new party to the call. The default is auto.

call

[-t {audio|video}]	Specify to add the participant as an audio or video call. The default is `video`.
<number>	Specify the IP address, or URI of the party to dial.

Examples:

```
set prompt "% "

ok,00
% control call dial 10.10.11.155

ok,00
CS,1,2,Ringing,Video,Normal,10.10.11.10,LifeSize
% control call dial -V -p h323 -t video 10.10.11.11 -b 1024

ok
CS,1,2,Ringing,Video,Normal,10.10.11.10,LifeSize
```

Dial using local directory entry:

```
% control call dial "local:john doe"

ok,00
CS,3,1,Ringing,Video,Normal,192.168.168.203,John Doe
%
```

Refer to "Asynchronous Messages" on page 212 for a description of the response.

display

The **display** target controls what remote source is shown on the local display.

Arguments:

[-s {main|aux| doc|auxdoc| playback|6|7| 8|9|10|11|12| 13|14|15}]	Specify the far camera source; the default is `main`. The description for these options is the same as the description for the output that appears in even-numbered columns 4 and greater in the video capabilities and far camera control messages. Refer to "Video Capabilities Messages" on page 218 or "Far Camera Control Messages" on page 216.
<callhandle>	Specify the call whose camera source is to be changed.

Examples:

control call display *4*   # Switch to the main video source

```
ok,00
```

control call display -s *doc 4*   # Switch to the document camera source

```
ok,00
```

dtmf

The **dtmf** target allows sending DTMF tones inband in an active call (for example, to access remote menu systems).

Arguments

`<callHandle>`	Specify the handle of the call to which to send digits.
`<{0-9｜A-D｜a-d｜*｜#}>`	Specify the digits to dial. The digits may be strung together for example, 5551212*#).

Examples:

control call dtmf *1 123456789abcd*#ABCD*

```
ok,00
```

hangup

The **hangup** target disconnects from a call. This command produces only asynchronous messages as a response to ensure proper ordering of displayed call status.

Arguments:

`[-a]`	Specify all active calls be terminated (cannot be used with `<handle>`).
`<handle>`	Specify the call or conference handle to disconnect.

Examples:

```
set prompt "% "

ok,00
% control call hangup 1

ok,00
CS,1,2,Terminated,Video,Normal,10.10.11.10,LifeSize

% control call hangup -a

ok,00
CS,1,2,Terminated,Video,Normal,10.10.11.10,LifeSize
```

Refer to "Asynchronous Messages" on page 212 for a description of the response.

> *Note:* The **hangup** command may occasionally report an error even though the call was actually hung up. To ensure the correct response for this command, refer to the asynchronous output that follows the command to determine the actual state of the command.

reboot

The **reboot** target causes the system to reboot.

Arguments:

`[seconds]`	Delay the reboot for the indicated number of seconds.

Examples:

```
control reboot 60

ok,00
```

> *Note:* Session terminated after 60 seconds.

record

The **record** object initiates and ends recordings.

The following targets apply to the **record** object.

start

The **start** target initiates recording with the specified recording key. If no key is supplied, the default key for the LifeSize video communications system is used.

Arguments:

[*record_key*]	Specify the recording key to use for the recording.

Examples

```
control record start 5417
```

```
ok,00
```

stop

The **stop** target ends a recording in progress.

Arguments:

None.

Examples

```
control record stop
```

```
ok,00
```

remote

The **remote** target emulates the silver remote control by sending sequences of commands that replicate the functionality of the remote control. The silver remote control is discontinued but still supported.

Arguments:

[-d msec]	Specifies the delay between press and release events in milliseconds. The default is 250 ms. This allows setting the hold down delay for all buttons in a given sequence. Minimum 0 ms, maximum 2000 ms. Cannot be used with -p or -r.
[-p]	Only send a key press event. Cannot be used with -d or -r or multiple buttons.
[-r]	Only send a key release event. Cannot be used with -d or -p or multiple buttons.
`<call|tri|squ|cir|back|ok|left|right|up|down|vup|vdn|mute|zin|zout|near|far|1|2|3|4|5|6|7|8|9|0|*|#|home|dir|yellow|red|blue|green>`	Specify the specific button to press. You can specify as many buttons on the command line as desired. Buttons are processed in the sequence given on the command line.

Note: The **home** and **dir** arguments do not correspond to any keys that are physically present on the remote, but serve as an aid to creating deterministic automated remote control sequences. The **home** argument goes to the main screen of the user interface. The **dir** argument goes to the directory screen from any other screen within the user interface.

Examples:

control remote left left ok zin 1 2 3 4 5 #

ok,00

Remote emulation software usage – user presses and holds a button:

control remote -p *left*

ok,00

User releases button:

control remote -r *left*

ok,00

remote1

The **remote1** target emulates the black remote control by sending sequences of commands that replicate the functionality of the remote control.

Arguments:

`[-d msec]`	Specifies the delay between press and release events in milliseconds. The default is 250 ms. This allows setting the hold down delay for all buttons in a given sequence. Minimum 0 ms, maximum 2000 ms. Cannot be used with `-p` or `-r`.
`[-p]`	Only send a key press event. Cannot be used with `-d` or `-r` or multiple buttons.
`[-r]`	Only send a key release event. Cannot be used with `-d` or `-p` or multiple buttons.

`<call|hup|tri|squ|`
`cir|back|ok|left|`
`right|up|down|vup|`
`vdn|mute|zin|zout|`
`nf|layout|input|`
`mode||1|2|3|4|`
`5|6|7|8|9|0|*|#|`
`home|dir|yellow|`
`red|blue|green>`
	Specify the specific button to press. You can specify as many buttons on the command line as desired. Buttons are processed in the sequence given on the command line. The `nf` argument refers to the button labeled **near/far** on the remote control.The `layout` and `mode` arguments refer to the ⬛ and ⌘ buttons respectively.

Note: The **dir** argument does not correspond to any key that is physically present on the remote, but serves as an aid to creating deterministic automated remote control sequences. The **dir** argument goes to the directory screen from any other screen within the user interface.

Examples:

control remote1 left left ok zin 1 2 3 4 5 #

ok,00

Remote emulation software usage: User presses and holds a button:

control remote1 -p *left*

ok,00

User releases button:

control remote1 -r *left*

ok,00

reset

The **reset** target configures the system to return to default configuration settings after the next system reboot.

Note: The reset target does not reboot the system. To return the system to default configuration settings after entering **control reset**, enter **control reboot**.

Arguments:

None

Examples:

control reset

ok,00

sleep

The **sleep** target puts the system into sleep mode.

Arguments:

None

Examples:

control sleep

ok,00

wakeup

The **wakeup** target wakes the system up from the sleep state if it was previously sleeping. It has no effect if the system is already awake.

Arguments:

>   None

Examples:

>   **control wakeup**
>
>   ok,00