Cisco Virtual Security Gateway Command-Line Interface

This chapter describes the Cisco Virtual Security Gateway (VSG) command-line interface (CLI).

This chapter includes the following sections:

Note Information about the Cisco VSG CLI is provided in this chapter. For information about the Cisco Nexus 1000V Series switch CLI or the Cisco Nexus 1010 Virtual Services Appliance CLI, see the respective product's documentation.

Information About the CLI Prompt

Once you have successfully accessed the system, the CLI prompt displays in the terminal window of your console port or remote workstation, as follows:

```
switch#
```

You can change this switch prompt to another name or leave it as it is.

```
switch# configure
switch(config)# switchname vsg100
switch(config)# exit
vsg100#
```

From the CLI prompt, you can do the following:

- Use CLI commands for configuring features.
- Access the command history.

- Use command parsing functions.

Command Modes

This section includes the following topics:

Information About Command Modes

The CLI is divided into command modes that define the actions available to the user. Command modes are "nested" and are accessed in sequence. When you first log in, you are placed in CLI EXEC mode.

As you navigate from EXEC mode to global configuration mode, a larger set of commands is available to you. To transition to global configuration mode, enter the following command:

config t

Table 2-1 shows how command access builds from user EXEC to global configuration mode.

Table 2-1   Accessing the Global Configuration Mode

Command Mode	Prompt	Description
EXEC	`vsg#`	- Connect to remote devices. - Temporarily change terminal line settings. - Do basic tests. - List system information (show).
Global configuration	`vsg(config)#`	Includes access to EXEC commands. - Connect to remote devices. - Temporarily change terminal line settings. - Perform basic tests. - List system information (show).

All commands in EXEC command mode are accessible from the global configuration command mode. For example, the **show** commands are available from any command mode.

EXEC Command Mode

When you first log in, you are placed into EXEC mode. The commands available in EXEC mode include the **show** commands that display device status and configuration information, the **clear** commands, and other commands that perform actions that you do not save in the device configuration.

Global Configuration Command Mode

Global configuration mode provides access to the widest range of commands, including those commands used to make configuration changes that are saved by the device and can be stored and applied when the device is rebooted.

Commands entered in global configuration mode update the running configuration file as soon as they are entered, but must also be saved into the startup configuration file by using the following command:

copy running-config startup-config

In global configuration mode, you can access protocol-specific, platform-specific, and feature-specific configuration modes.

Exiting a Configuration Mode

To exit from any configuration mode, use one of the following commands:

Command	Purpose

exit
 	
Example:
`vsg(config-rule)# exit`
`vsg(config)#`
	Exits from the current configuration command mode and returns to the previous configuration command mode.

end
 	
Example:
`vsg(config)# end`
`vsg#`
	Exits from the configuration command mode and returns to EXEC mode.

Ctrl-z
 	
Example:
`vsg(config)# ^z`
`vsg#`
	Exits the current configuration command mode and returns to EXEC mode.
	
	⚠
	Caution If you press **Ctrl-Z** at the end of a command line in which a valid command has been typed, the CLI adds the command to the running configuration file. We recommend that you exit a configuration mode using the **exit** or **end** command.

Command Mode Summary

Table 2-2 summarizes information about command modes.

Table 2-2   *Command Mode Summary*

Mode	Access Method	Prompt	Exit Method
EXEC	From the login prompt, enter your username and password.	`VSG#`	To exit to the login prompt, use the **exit** command.
Global configuration	From EXEC mode, enter the command, **configure**.	`VSG(config)#`	To exit to EXEC mode, use the **end** or **exit** command or press **Ctrl-Z**.
Zone configuration	From global configuration mode, enter the command, **zone** *zone-name*.	`VSG(config-zone)#`	To exit to global configuration mode, use the **exit** command. To exit to EXEC mode, use the **end** command or press **Ctrl-Z**.
Data0 interface configuration	From global configuration mode, enter the command **interface data0**	`VSG(config-if)#`	To exit to global configuration mode, use the **exit** command. To exit to EXEC mode, use the **end** command or press **Ctrl-Z**.

Saving CLI Configuration Changes

This section describes how to save CLI configuration changes and includes the following topics:

Running Configuration

The running configuration is the configuration that is currently running on the device. It includes configuration changes from commands entered since the last time the device was restarted. If the device is restarted, the running configuration is replaced with a copy of the startup configuration. Any changes that were made to the running configuration but were not copied to the startup configuration are discarded.

Startup Configuration

The startup configuration is the configuration that is saved and that will be used by the device when you restart it. When you make configuration changes to the device, they are automatically saved in the running configuration. If you want configuration changes saved permanently, you must copy them to the startup configuration so that they are preserved when the device is rebooted or restarted.

Copying the Running Configuration to the Startup Configuration

To copy changes you have made to the running configuration into the startup configuration so that they are saved persistently through reboots and restarts, use the following command:

	Command	Purpose

Step 1	`copy running-config startup-config`
	
	Example:
	`vsg(config)# copy running-config startup-config`
		(Optional) Saves the running configuration persistently through reboots and restarts by copying it to the startup configuration.

Special Characters

Table 2-3 lists the characters that have special meaning in text strings and should be used only in regular expressions or other special contexts.

Table 2-3   Special Characters

Character	Description
|	Vertical bar
< >	Less than or greater than

Keystroke Shortcuts

Table 2-4 lists command key combinations that can be used in both EXEC and configuration modes.

Table 2-4   Keystroke Shortcuts

Key(s)	Description
Ctrl-A	Moves the cursor to the beginning of the line.
Ctrl-B	Moves the cursor one character to the left. When you enter a command that extends beyond a single line, you can press the Left Arrow or Ctrl-B keys repeatedly to scroll back toward the system prompt and verify the beginning of the command entry, or you can press the Ctrl-A key combination.
Ctrl-C	Cancels the command and returns to the command prompt.
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Moves the cursor to the end of the line.
Ctrl-F	Moves the cursor one character to the right.
Ctrl-G	Exits to the previous command mode without removing the command string.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-L	Redisplays the current command line.
Ctrl-R	Redisplays the current command line.

Table 2-4   Keystroke Shortcuts (continued)

Key(s)	Description
Ctrl-T	Transposes the character to the left of the cursor with the character located to the right of the cursor.
Ctrl-U	Deletes all characters from the cursor to the beginning of the command line.
Ctrl-W	Deletes the word to the left of the cursor.
Ctrl-X, H	Lists history. When using this key combination, press and release the Ctrl and X keys together before pressing H.
Ctrl-Y	Recalls the most recent entry in the buffer (press keys simultaneously).
Ctrl-Z	Ends a configuration session, and returns you to EXEC mode. When used at the end of a command line in which a valid command has been typed, the resulting configuration is first added to the running configuration file.
↑	Displays the previous command in the command history.
↓	Displays the next command in the command history.
→ ←	Moves your cursor through the command history directionally to locate a command string.
?	Displays a list of available commands.

Tab	Completes the word for you after you enter the first characters of the word and then press the Tab key. All options that match are presented.
	
	Used to complete:
	• Command names
	• Scheme names in the file system
	• Server names in the file system
	• File names in the file system
	
	This example shows how to use the tab keystroke:
	
	`vsg(config)# xm<Tab>`
	`vsg(config)# xml <Tab>`
	`vsg(config)# xml server`
	
	This example shows how to use the tab keystroke:
	
	`vsg(config)# vn<Tab>`
	`vnm-policy-agent vns-binding`
	`vsg(config)# security-pr<Tab>`
	`vsg(config)# security-profile`

Abbreviating Commands

You can abbreviate commands and keywords by entering the first few characters of a command. The abbreviation must include enough characters to make it unique from other commands or keywords. If you are having trouble entering a command, check the system prompt and enter the question mark (?) for a list of available commands. You might be in the wrong command mode or using incorrect syntax.

Table 2-5 lists examples of command abbreviations.

Table 2-5   Examples of Command Abbreviations

Command	Abbreviation
configure	**conf**
copy running-config startup-config	**copy run start**
show running-config	**sho run**

Using the no Form of a Command

Almost every configuration command has a **no** form that can be used to disable a feature or function. For example, to remove a VLAN, use the **no vlan** command. To re-enable it, use the **vlan** command form.

For example, if you use the **boot** command in global configuration mode, you can then use the **no boot** command to undo the results:

```
vsg(config)# boot system bootflash: svs1.bin
vsg(config)# no boot system bootflash: svs1.bin
```

Using Help

The CLI provides the following help features (see Table 2-6 and Table 2-7).

Table 2-6   CLI Help Features

Feature	Description
?	Type the question mark (?) to list the valid input options.
^	The CLI prints the caret (^) symbol below a line of syntax to point to an input error in the command string, keyword, or argument.
↑	Use the up arrow to have the CLI display the previous command you entered so that you can correct an error.

The example in Table 2-7 describes how to use syntax error isolation and context-sensitive help.

Table 2-7   *Using Syntax Error Isolation and Context-Sensitive Help on the CLI*

	Command	Purpose

Step 1	`show interface ?`
	
	Example:
	`vsg# show interface ?`
	`  <CR>`
	`  >                 Redirect it to a file`
	`  >>                Redirect it to a file in append mode`
	`  brief             Show brief info of interface`
	`  capabilities      Show interface capabilities information`
	`  counters          Show interface counters`
	`  data              Data interface`
	`  debounce          Show interface debounce time information`
	`  description       Show interface description`
	`  ethernet          Ethernet IEEE 802.3z`
	`  fcoe (no abbrev)  Show FCoE info for interface`
	`  loopback          Loopback interface`
	`  mac-address       Show interface MAC address`
	`  mgmt              Management interface`
	`  port-channel      Port Channel interface`
	`  snmp-ifindex      Show snmp ifindex list`
	`  status            Show interface line status`
	`  switchport        Show interface switchport information`
	`  transceiver       Show interface transceiver information`
	`  trunk             Show interface trunk information`
	`  vethernet         Virtual ethernet interface`
	`  virtual           Show virtual interface information`
	`  |                 Pipe command output to filter`
	
	`vsg#`
		Displays the optional parameters used with the **show interface** command in EXEC mode.

Step 2	`show interface module ?`
	
	Example:
	`vsg# show interface module ?`
	`                    ^`
	`Invalid command (interface name) at '^' marker.`
	`?`
	`vsg#`
		Displays an invalid command error message and points (^) to the syntax error.

Step 3	`Ctrl-P` or the Up Arrow
	
	Example:
	`vsg# <Ctrl-P>`
	`vsg# show interface data0`
		Displays the previous command you entered so that you can correct the error.

Step 4	`show interface data ?`
	
	Example:
	`vsg# show interface data ?`
	`  <0-0>  Data interface number`
	`vsg#`
		Displays the syntax for showing a data interface (data0).

Table 2-7   Using Syntax Error Isolation and Context-Sensitive Help on the CLI (continued)

	Command	Purpose

Step 5	`show interface data0`
	
	Example:
	`vsg# show interface data0`
	`control0 is up`
	`  Hardware: Ethernet, address: 0050.5691.53b6 (bia`
	`0050.5691.53b6)`
	`  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,`
	`    reliability 255/255, txload 1/255, rxload 1/255`
	`  Encapsulation ARPA`
	`  full-duplex, 1000 Mb/s`
	`  Auto-Negotiation is turned on`
	`  1 minute input rate 1920 bits/sec, 0 packets/sec`
	`  1 minute output rate 24 bits/sec, 0 packets/sec`
	`  Rx`
	`    91082 input packets 0 unicast packets 2935 multicast`
	`packets`
	`    88147 broadcast packets 20642956 bytes`
	`  Tx`
	`    21968 output packets 0 unicast packets 21968 multicast`
	`packets`
	`    0 broadcast packets 5228289 bytes`
	
	`vsg#`
		Displays the data interface (data0).