



CHAPTER 1

Command-Line Interface

Revised: September 27, 2012, OL-24149-05

Introduction

This chapter describes how to use the SCE platform Command-Line Interface (CLI), its hierarchical structure, authorization levels and its help features. The Command-Line Interface is one of the SCE platform management interfaces.

The CLI is accessed through a Telnet session or directly via the console port on the front panel of the SCE platform. When you enter a Telnet session, you enter as the simplest level of user, in the User Exec mode.

The SCE platform supports up to eleven concurrent CLI sessions; five sessions initiated by Telnet connection, five sessions by SSH connection, and one session on the console port.

- [Authorization and Command Levels \(Hierarchy\), page 1-2](#)
- [CLI Help Features, page 1-10](#)
- [Navigational and Shortcut Features, page 1-12](#)
- [Managing Command Output, page 1-15](#)
- [Creating a CLI Script, page 1-17](#)

Authorization and Command Levels (Hierarchy)

- [CLI Command Mode Hierarchy, page 1-3](#)
- [Prompt Indications, page 1-6](#)
- [Navigating Between Authorization Levels and Command Modes, page 1-7](#)

When using the CLI there are two important concepts that you must understand to navigate:

- **Authorization Level** — Indicates the level of commands you can execute. A user with a simple authorization level can only view some information in the system, while a higher level administrator can actually make changes to configuration.

This manual documents commands at the user exec, privileged exec, and admin authorization levels.

- **Command Hierarchy Level** — Provides you with a context for initiating commands. Commands are broken down into categories and you can only execute each command within the context of its category. For example, to configure parameters related to the Line Card, you need to be within the Linecard Interface Configuration Mode. [CLI Command Mode Hierarchy, page 1-3](#)

The following sections describe the available Authorization and Command Hierarchy Levels and how to maneuver within them.

The on-screen prompt indicates both your authorization level and your command hierarchy level, as well as the assigned host name.

**Note**

Throughout the manual, SCE is used as the sample host name.

CLI Authorization Levels

The SCE platform has four authorization levels, which represent the user access permissions. When you initially connect to the SCE platform, you automatically have the most basic authorization level, that is User, which allows minimum functionality.

To monitor the system, you must have Viewer authorization, while to perform administrative functions on the SCE platform, you must have Admin or Root authorization. A higher level of authorization is accessed by logging in with appropriate password, as described in the procedures below.

In each authorization level, all the commands of the lower authorization layers are available in addition to commands that are authorized only to the current level.

The following CLI commands are related to authorization levels:

- **enable**
- **disable**

Each authorization level has a value (number) corresponding to it. When using the CLI commands, use the values, not the name of the level, as shown in [Table 1-1](#).

Table 1-1 Authorization Levels

Level	Description	Value	Prompt
User	Password required. This level enables basic operational functionality.	0	>
Viewer	Password required. This level enables monitoring functionality. All show commands are available to the Viewer authorization level, with the exception of those that display password information.	5	>
Admin	Password required. For use by general administrators, the Admin authorization level enables configuration and management of the SCE platform.	10	#
Root	Password required. For use by technical field engineers, the Root authorization level enables configuration of all advanced settings, such as debug and disaster recovery. The Root level is used by technical engineers only.	15	#>

CLI Command Mode Hierarchy

The set of all CLI commands is grouped in hierarchical order, according to the type of the commands. The first two levels in the hierarchy are the User Exec and Privileged Exec modes. These are non-configuration modes in which the set of available commands enables the monitoring of the SCE platform, file system operations, and other operations that cannot alter the configuration of the SCE platform.

The next levels in the hierarchy are the Global and Interface configuration modes, which hold a set of commands that control the global configuration of the SCE platform and its interfaces. Any of the parameters set by the commands in these modes should be saved in the startup configuration, such that in the case of a reboot, the SCE platform restores the saved configuration.

[Table 1-2](#) shows the available CLI modes.

Table 1-2 CLI Modes

Mode	Description	Level	Prompt indication
User Exec	Initial mode. Also allows monitoring of the system (show commands).	User/Viewer	SCE >
Privileged Exec	General administration; file system manipulations and control of basic parameters that do not change the configuration of the SCE platform.	<ul style="list-style-type: none"> • Admin • Root 	<ul style="list-style-type: none"> • SCE # • SCE #>
Global Configuration	Configuration of general system parameters, such as DNS, host name, and time zone.	<ul style="list-style-type: none"> • Admin • Root 	<ul style="list-style-type: none"> • SCE (config)# • SCE (config)# >

Table 1-2 *CLI Modes (continued)*

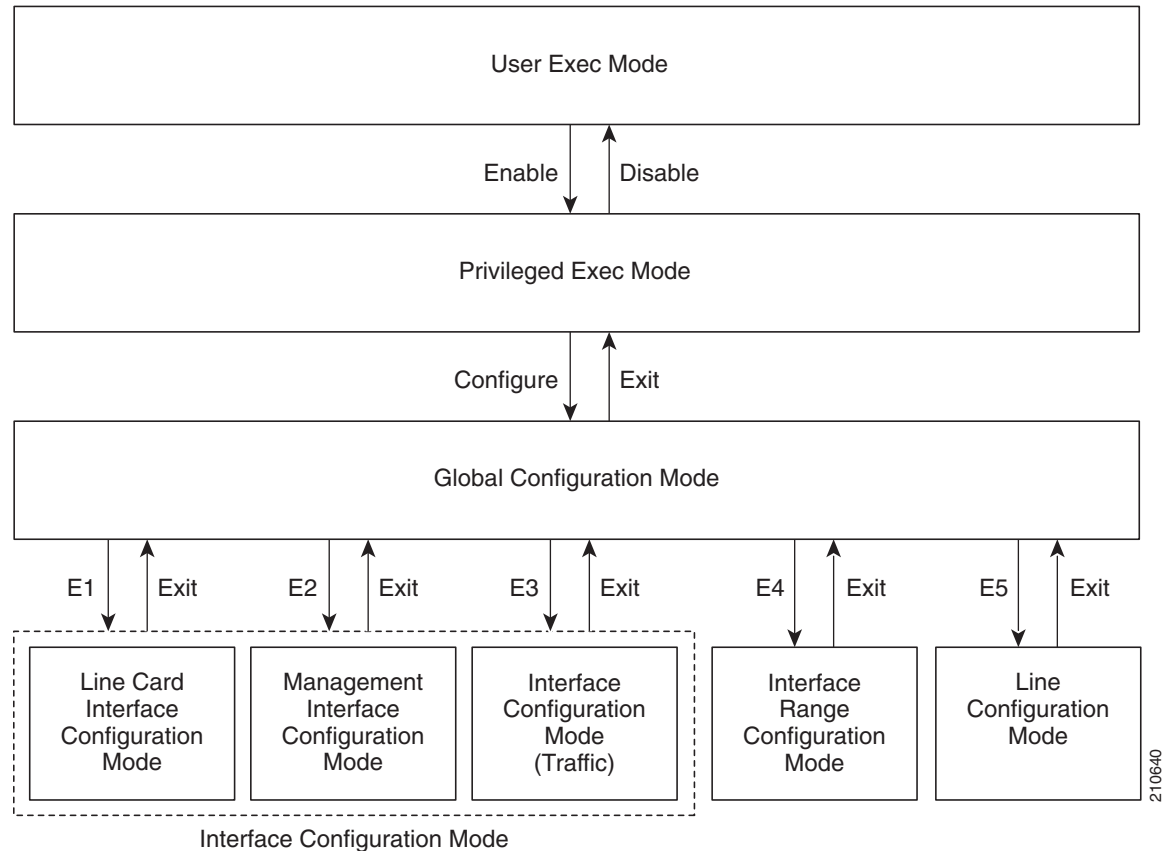
Mode	Description	Level	Prompt indication
Interface Configuration	Configuration of specific system interface parameters, for the following interface modes. <ul style="list-style-type: none"> • linecard interface • management interface • specific traffic interface 	<ul style="list-style-type: none"> • Admin • Root 	<ul style="list-style-type: none"> • SCE(config if)# • SCE(config if)#>
Interface Range Configuration	Configuration of a range of traffic interfaces.	<ul style="list-style-type: none"> • Admin • Root 	<ul style="list-style-type: none"> • SCE(config if range)# • SCE(config if range)#>
Line Configuration	Configuration of Telnet lines, such as an access-list.	<ul style="list-style-type: none"> • Admin • Root 	<ul style="list-style-type: none"> • SCE (config-line)# • SCE (config-line)#>

When you login to the system, you have the User authorization level and enter User Exec mode. Changing the authorization level to Viewer does not change the mode. Changing the authorization level to Admin automatically moves you to Privileged Exec mode. To move to any of the configuration modes, you must enter commands specific to that mode.

The list of available commands in each mode can be viewed using the question mark '?' at the end of the prompt.

Figure 1-1 illustrates the hierarchical structure of the CLI modes, and the CLI commands used to enter and exit a mode.

Figure 1-1 CLI Command Modes



The following commands are used to enter the different configure interface modes and the Line Configuration Mode:

- E1 **interface Linecard 0**
- E2 **interface Mng 0/1** or **0/2** (management port, all platforms)
- E3:
 - SCE 1000: **interface GigabitEthernet 0/1** or **0/2**
 - SCE 2000: **interface TenGigabitEthernet 0/1, 0/2, 0/3, or 0/4**
- E4:
 - SCE 1000: **interface range GigabitEthernet 0/1-2**
 - SCE 2000: **interface range TenGigabitEthernet 0/<port-range (any range between 1 and 4)>**
- E5 **line vty 0**

**Note**

Although the system supports up to five concurrent Telnet connections, you cannot configure them separately. This means that any number you enter in the **line vty** command (**0, 1, 2, 3** or **4**) will act as a **0** and configure all five connections together.

**Note**

In order for the auto-completion feature to work, when you move from one interface configuration mode to another, you must first exit the current interface configuration mode (as illustrated in the above figure).

Example:

This example illustrates moving into and out of configuration modes as follows:

- Enter global configuration mode
- Configure the SCE platform time zone
- Enter Mng Interface configuration mode for Mng port 1
- Configure the speed of the management interface
- Exit the Mng Interface configuration mode to the global configuration mode
- Enter the Linecard Interface configuration
- Define the link mode
- Exit Linecard Interface configuration mode to the global configuration mode
- Exit global configuration mode

```
SCE#configure
SCE(config)#clock timezone PST -10
SCE(config)#interface Mng 0/1
SCE(config if)#speed 100
SCE(config if)#exit
SCE(config)#interface Linecard 0
SCE(config if)#link mode all-links forwarding
SCE(config if)#exit
SCE(config)#exit
sce#
```

Prompt Indications

The on-screen prompt indicates your authorization level, your command hierarchy level, and the assigned host name. The structure of the prompt is:

<hostname (mode-indication) level-indication>

Authorization levels are indicated as listed in [Table 1-3](#).

Table 1-3 *Prompt Indications: Authorization levels*

This prompt...	Indicates this...
>	indicates User and Viewer levels
#	indicates Admin level
#>	indicates Root level

Command modes are indicated as listed in [Table 1-4](#).

Table 1-4 Prompt Indications: Command Modes

This command hierarchy...	Is indicated as...
User Exec	SCE>
Privileged Exec	sce#
Global Configuration	SCE (config)#
Interface Configuration	SCE (config if)#
Interface Range Configuration	SCE (config if range)#
Line Configuration	SCE (config-line)#

Example:

The prompt `SCE1(config if)#` indicates:

- The name of the SCE platform is `SCE1`
- The current CLI mode is Interface configuration mode
- The user has Admin authorization level

Navigating Between Authorization Levels and Command Modes

The authorization levels and command modes function together in one hierarchy. The User and Viewer authorization levels have only a single command mode. When you enter either the Admin or Root authorization level (which function in parallel), you enter the Privileged Exec command mode. From this command mode you can access the other command modes.

- User Exec authorization level
- Viewer authorization level
- Privileged Exec command mode (you are now in either Admin or Root authorization level)
- Global Configuration command mode

From this command mode, the following Interface Command Modes can be accessed:

- Management Interface Configuration
- Linecard Interface Configuration
- GigabitEthernet Interface Configuration (GBE traffic interfaces)
- Interface Range Configuration (range of traffic interfaces)
- Line Configuration

Table 1-5 summarizes how to navigate the CLI command hierarchy.

Table 1-5 CLI Command Hierarchy

Authorization Level or Command Mode	Use this command to access	Use this command to exit
User Exec	Not applicable	logout or exit (exits the current CLI session)
Viewer	enable 5	disable
Privileged Exec	enable 10 or enable 15 (accesses root level)	disable
Global Configuration	configure	exit (exits to Privileged Exec) end (exits to User Exec)
Management Interface Configuration (management)	interface Mng 0/1 or 0/2	exit (exits to Global Configuration) end (exits to User Exec)
Linecard Interface Configuration	interface linecard 0	exit (exits to Global Configuration) end (exits to User Exec)
GigabitEthernet Interface Configuration (traffic)	SCE 2000: interface gigabitethernet 0/1, 0/2, 0/3, or 0/4 SCE 1000: interface gigabitethernet 0/1 or 0/2 OR SCE 2000: interface range gigabitethernet 0/<port-range (any range between 1 and 4) SCE 1000: interface range gigabitethernet 0/1-2	exit (exits to Global Configuration) end (exits to User Exec)
Line Configuration	line vty 0	exit (exits to Global Configuration) end (exits to User Exec)

Configuring the Physical Ports

The SCE platform contains the following physical port interfaces:

- Management:

The following commands are used to configure the management port:

- ip address
- duplex
- speed
- active-port
- auto-fail-over

- Gigabit Ethernet

The following commands are used to configure the Gigabit Ethernet line ports. These are the commands for which the range interface mode is relevant; use the **range interface** command to configure more than one interface at a time if you are configuring them to identical values.

- auto-negotiate
- bandwidth
- queue

CLI Help Features

CLI provides context sensitive help. Two types of context sensitive help are supported:

- [Partial Help, page 1-10](#)
- [Argument Help, page 1-10](#)

Partial Help

To obtain a list of commands that begin with a particular character string, enter the abbreviated command entry immediately followed by a question mark (?). This form of help is called partial help, because it lists only the keywords or arguments that begin with the abbreviation you entered.

Example:

The following example illustrates how typing **c?** displays all available arguments that start with the letter **c**.

```
SCE(config)#snmp-server c?
Community contact
SCE(config)#snmp-server c
```

Argument Help

To obtain a list of command's associated keywords or parameters, type a question mark (?) in place of a keyword or parameter on the command line.

Note that if **<Enter>** is acceptable input, the symbol **<cr>** represents the **Enter** key.

Example:

The following example illustrates how to get a list of all arguments or keywords expected after the command **snmp-server**.

```
SCE(config)#snmp-server ?
community Define community string
contact Set system contact
enable Enable the SNMP agent
host Set traps destination
interface Set interface parameters
SCE(config)# snmp-server
```

When asking for help on particular parameter, the system informs you of the type of data that is an accepted legal value. The types of parameters supported are:

- STRING:** When a String is expected, you can enter any set of characters or digits. If the string has a space as one of its characters, use double-quote (") marks to enclose the string.
- DECIMAL:** Any decimal number. Positive number is assumed, for negative numbers use the "-" symbol.
- HEX:** A hexadecimal number; must start with either 0x or 0X.

Example:

The following example illustrates the use of ? to get help on commands syntax. In this example, you can enter either the word **running-config**, or any name of a file, after the word **copy**.

```
SCE#copy ?
running-config Copy running configuration file
startup-config Backup the startup-config to a specified destination
STRING          Source file
SCE#copy
```

Table 1-6 summarizes the CLI help features.

Table 1-6 Getting Help

Command	Purpose
?	List all commands available for a particular command mode
<abbreviated-command-entry>? Example: c? calendar cd clear clock configure copy copy-passive	Obtain a list of commands that begin with a particular character string. (Do not leave a space between the command and question mark.)
<abbreviated-command-entry><Tab>	Complete a partial command name.
<command>?	List the keywords associated with the specified command.
<command keyword> ? Example: show ? access-lists Show all access-lists	List the arguments associated with the specified keyword. Leave a space between the keyword and question mark

Navigational and Shortcut Features

- [Command History, page 1-12](#)
- [Keyboard Shortcuts, page 1-12](#)
- [Auto-Completion, page 1-13](#)
- [FTP User Name and Password, page 1-14](#)
- [The "do" Command: Executing Commands Without Exiting, page 1-14](#)

Command History

CLI maintains a history buffer of the most recent commands you used in the current CLI session for quick retrieval. Using the keyboard, you can navigate through your last commands, one by one, or all commands that start with a given prefix. By default, the system saves the last 30 commands you typed. You can change the number of commands remembered using the **history size** command.

To use the history functions, use the keys shown in [Table 1-7](#).

Table 1-7 Keyboard Shortcuts for History Functions

Arrow	Shortcut	Description
Up arrow	Ctrl-P	Move cursor to the previous command with the same prefix.
Down arrow	Ctrl-N	Moves the cursor to the next command with the same prefix as original.
	Ctrl-L	Re-display the current command line.
	Ctrl-R	

Keyboard Shortcuts

The SCE platform has several keyboard shortcuts that make it easier to navigate and use the system. [Table 1-8](#) shows the keyboard shortcuts available.

You can get a display the keyboard shortcuts at any time by typing **help bindings**.

Table 1-8 Keyboard Shortcuts

Description	Shortcut key
Navigational shortcuts	
Move cursor one character to the right.	CTRL-F /->
Move cursor one character to the left.	CTRL-B /<-
Move cursor one word to the right (forward).	ESC-F
Move cursor one word to the left (backward).	ESC-B
Move cursor to the start of the line.	CTRL-A
Move cursor to the end of the line.	CTRL-E
Editing shortcuts	

Table 1-8 Keyboard Shortcuts (continued)

Description	Shortcut key
Delete the character where the cursor is located.	CTRL-D
Delete from the cursor position to the end of the word.	ESC-d
Delete the character before the current location of the cursor.	Backspace
Delete the character before the current location of the cursor.	CTRL-H
Deletes from the cursor position to the end of the line	CTRL-K
Deletes all characters from the cursor to the beginning of the line	CTRL-U
Delete the word to the left of the cursor.	CTRL-W
Recall the last item deleted.	CTRL-Y
Completes the word when there is only one possible completion.	<Tab>
Completes the word when there is only one possible completion. (Same functionality as <Tab>.)	CTRL-I

Auto-Completion

The CLI interface features tab completion. When you type in the first letters of a command and type **<Tab>**, the system automatically fills in the rest of the command or keyword. This feature works only when there is one command that could be possible using the starting letters.

Example 1

The letters **snm** followed by **<Tab>** will be completed to the command **snmp-server**.

```
SCE(config)#snm <Tab>
SCE(config)#snmp-server
```

If you type **<Enter>** instead of **<Tab>**, and there is no ambiguity, the system actually carries out the command that is the result of the auto-completion.

Example 2

The following example displays how the system completes a partial (unique) command for the **enable** command. The system carries out the command using the default authorization level (10) when you press **Enter**.

```
SCE>en <Enter>
Password:
sce#
```

Example 3

The following example illustrates how to use the completion feature with a non-default value for the argument. In this example, the **enable** command is completed using the specified value (15) for the authorization level.

```
SCE>en 15 <Enter>
Password:
sce#
```

FTP User Name and Password

CLI enables saving FTP user name and password to be used in FTP operations—download and upload, per session.

These settings are effective during the current CLI session.

The following example illustrates how to set FTP password and user name and the use in these settings for getting a file named *config.tmp* from a remote station using FTP protocol.

```
sce#ip FTP password 1234
sce#ip FTP username cisco
sce#copy ftp://@10.10.10.10/h:/config.tmp myconf.txt
connecting 10.1.1.253 (user name cisco password 1234) to retrieve config.tmp
sce#
```

The "do" Command: Executing Commands Without Exiting

There are four configuration command modes:

- Global configuration mode
- Management interface configuration mode
- Interface configuration mode
- Line configuration mode

When you are in one of these configuration modes, it is possible to execute an EXEC mode command (such as a **show** command) or a privileged EXEC (such as **show running-config**) without exiting to the relevant command mode. Use the '**do**' command for this purpose.

How to execute an exec mode command from a configuration command mode

Step 1 At the SCE(config)# (or SCE(config if)#) prompt, type **do** *<command>*.

The specified command executes without exiting to the appropriate exec command mode.

The following example shows how to display the running configuration while in interface configuration mode.

```
SCE(config if#) do show running-config
```

Managing Command Output

- [Scrolling the Screen Display, page 1-15](#)
- [Filtering Command Output, page 1-15](#)
- [Redirecting Command Output to a File, page 1-16](#)

Some commands, such as many show commands, may have many lines of output. There are several ways of managing the command output:

- Scrolling options — When the command output is too large to be displayed all at once, you can control whether the display scrolls line by line or refreshes the entire screen.
- Filtering options — You can filter the output so that output lines are displayed only if they include or exclude a specified expression.
- Redirecting to a file — You can send the output to a specified file.

Note that by default, the show commands act the same as the more commands; that is, the output is displayed interactively a single screen at a time. Use the no more (on page) command to disable this feature so that show commands display the complete output all at one time.

Scrolling the Screen Display

The output of some **show** and **dir** commands is quite lengthy and cannot all be displayed on the screen at one time. Commands with many lines of output are displayed in chunks of 24 lines. You can choose to scroll the display line by line or refresh the entire screen. At the prompt after any line, you can type one of the following keys for the desired action:

- **<Enter>**- Show one more line
- **<Space>**- Show 24 more lines (a new chunk)
- **<g>**- Stop prompting for more
- **<?>**- Display a help string showing possible options
- Any other key- Quit showing the file

Filtering Command Output

You can filter the output of certain commands, such as **show**, **more**, and **dir**, so that output lines are displayed only if they include or exclude a specified expression. The filtering options are as follows:

- **include** — Shows all lines that include the specified text.
- **exclude** — Does not show any lines that include the specified text.
- **begin** — Finds the first line that includes the specified text, and shows all lines starting from that line. All previous lines are excluded.

The syntax of filtered commands is as follows:

- `<command>| include <expression>`
- `<command>| exclude <expression>`
- `<command>| begin <expression>`

Following is an example of how to filter the **show version** command to display only the last part of the output, beginning with the version information.

```
sce# show version | begin revision
```

Redirecting Command Output to a File

You can redirect the output of commands, such as **show**, **more**, and **dir**, to a file. When writing the output of these commands to a file, you can specify either of the following options:

- **redirect** — The new output of the command will overwrite the existing contents of the file.
- **append** — The new output of the command will be appended to the existing contents of the file.

The syntax of redirection commands is as follows:

- `<command>| redirect <file-name>`
- `<command>| append <file-name>`

Following is an example of how to do the following:

- Filter the **more** command to display from a csv subscriber file only the gold package subscribers.
- Redirect that output to a file named `current_gold_subscribers`. The output should not overwrite existing entries in the file, but should be appended to the end of the file.

```
sce# more subscribers_10.10.2004 include gold | append current_gold_subscribers
```


Creating a CLI Script

The CLI scripts feature allows you to record several CLI commands together as a script and play it back. This is useful for saving repeatable sequence of commands, such as software upgrade. For example, if you are configuring a group of SCE platforms and you want to run the same configuration commands on each platform, you could create a script on one platform and run it on all the other SCE platforms. The available script commands are:

- **script capture**
- **script stop**
- **script print**
- **script run**

-
- Step 1** At the `sce#` prompt, type **script capture *sample1.scr*** where *sample1.scr* is the name of the script.
- Step 2** Perform the actions you want to be included in the script.
- Step 3** Type **script stop**.
- The system saves the script.
-

The following is an example of recording a script for upgrading software.

```
SCE#script capture upgrade.scr
sce#configure
SCE(config)#boot system new.pkg
Verifying package file...
Package file verified OK.
SCE(config)#exit
sce#copy running-config startup-config
Writing general configuration file to temporary location...
Extracting files from '/tffs0/images/new.pkg'...
Verifying package file...
Package file verified OK.
Device '/tffs0/' has 81154048 bytes free, 21447973 bytes are needed for extraction, all is
well.
Extracting files to temp locations...
Renaming temp files...
Extracted OK.
Backing-up general configuration file...
Copy temporary file to final location...
sce#script stop
sce#
```

