



Using the Command Line Interface

The Command Line Interface (CLI) is a line-oriented user interface that has a set of commands for configuring, managing, and monitoring the CSS. To help you use these commands, this chapter provides you with information on:

- CLI Modes
- Logging into and Exiting the CLI
- Using CLI Commands
- Getting CLI Help
- User-Defined Variables
- CSS Scripts

For information on specific CLI commands, refer to Chapter 2, [CLI Commands](#).

CLI Modes

The CLI provides the following modes:

- User
- SuperUser
- Configuration and its subordinate modes

The following sections describe each of these modes.

User Mode

When the CSS administrator assigns a username with User permission, this permission level allows you to log in to User mode on the CSS. This mode is identified by a prompt that ends with a greater-than symbol (>). Within this mode, you can use a limited set of commands to monitor and display CSS parameters, but not change them.

For information on the commands you can use in User mode, refer to Chapter 2, [CLI Commands](#), “General Commands”.

SuperUser Mode

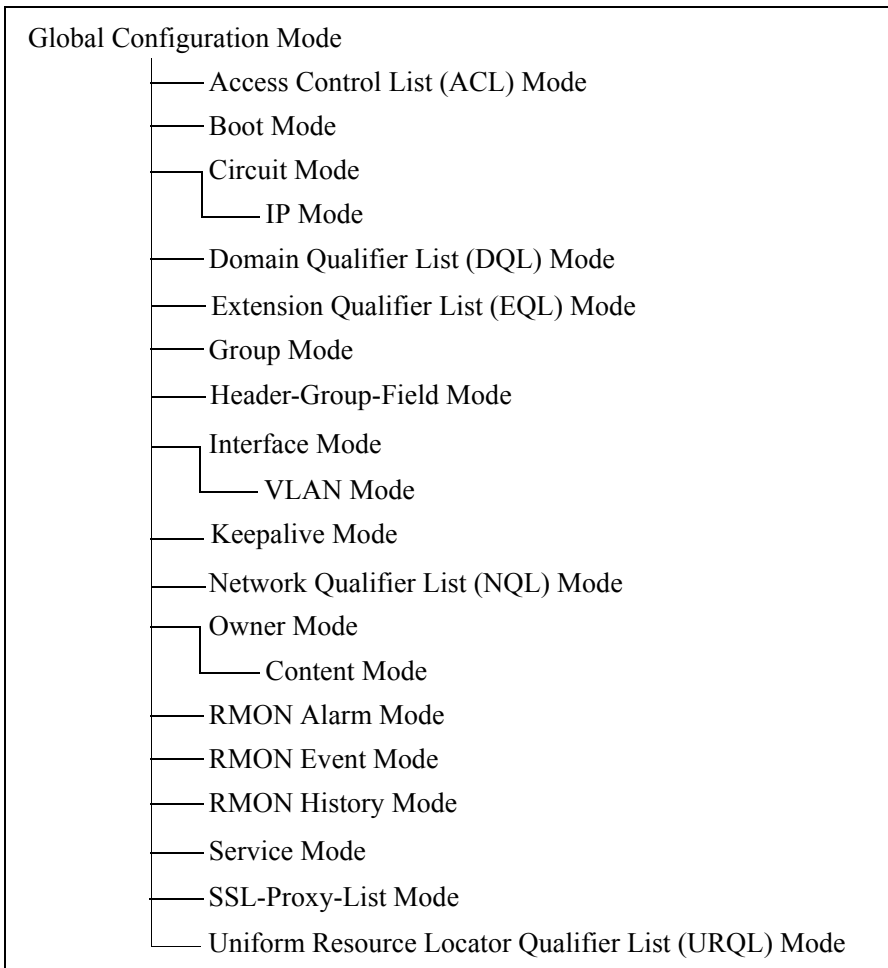
When the CSS administrator assigns a username with SuperUser permission, this permission level allows you to log in to SuperUser mode on the CSS. This mode is identified by a prompt that ends with a pound sign (#).

Within this mode, you can use the commands to maintain the CSS and to access global configuration mode to configure the CSS. You can also use User-mode commands in SuperUser mode.

For information on the commands you can use in SuperUser mode, refer to Chapter 2, [CLI Commands](#), “General Commands”.

Configuration Modes

When you log in to the CSS as a SuperUser, you can use the **configure** command to access global configuration mode. This mode is identified by a (config) prompt. Within this mode, you can use its set of commands to configure the CSS and access its subordinate configuration modes.



When you access any of the subordinate configuration modes, the CSS appends the mode name to the (config) prompt. For example, when you access owner mode from global configuration mode, the prompt changes to (config-owner [owner_name]).

Each mode has its own set of commands. Many of the modes have commands allowing you to access other related modes. When you switch modes, you exit the current mode and enter the new mode. For example, from interface configuration mode, you can move directly to ACL, circuit, DQL, EQL, group, owner, RMON alarm, RMON event, RMON history, service, or URQL configuration mode.

To exit any configuration mode and return to SuperUser mode, press CTRL-z.

For information on the commands you can use in global configuration mode and its subordinate modes including the **configure** command, refer to Chapter 2, [CLI Commands](#).

Logging into and Exiting the CLI

When the CSS completes the system boot, the CLI starts automatically and is available for use. To log in to the CSS and access the CLI, use a terminal device connected to the Console port on the CSS.

Instead of using the Console port, you can also use Telnet. For more information on terminal settings for Telnet use, refer to the *Cisco 11500 Series Content Services Switch Hardware Installation Guide* for the 11500 series CSS or the *Cisco Content Services Switch Getting Started Guide* for the 11000 series CSS.

You can use the CLI from any terminal device that is compatible with ANSI, VT52, or VT100 characteristics. ANSI and VT100 devices let you use cursor-control and cursor-movement keys: left-arrow, up-arrow, down-arrow, right-arrow, Delete, and Backspace. The CLI senses the use of cursor-control keys and uses the optimal device characteristics automatically.



Note

The first time you log in to the CLI, use the default username of **admin** and the default password of **system**.

To exit from the CLI, use the **exit** command from SuperUser mode. If you are currently in one of the configuration modes and want to exit the CLI, use CTRL-z to return to SuperUser mode and then use the **exit** command.

Using CLI Commands

This section provides information on:

- Command conventions for syntax used in this book and variable argument conventions
- Entering multiple commands
- CLI keyboard shortcuts when you are using the CLI
- Using `grep`
- Understanding CLI syntax checking and error messages

Syntax Conventions

To help you identify the parts of a CLI command, Chapter 2, [CLI Commands](#), uses conventions to show the syntax of commands. [Table 1-1](#) lists these syntax conventions and their descriptions.

Table 1-1 *Syntax Conventions*

Syntax Convention	Description
boldface	Identifies commands and options you must enter exactly as shown.
<i>italics</i>	Identifies variables you must supply. For more information on variable arguments, see the next section.
... (ellipsis)	Identifies the continuation of the command.
(vertical bar)	Identifies mutually exclusive choices. Note that you can also use the character as a pipe with grep . For more information on grep , see “Using grep” .
[] (square brackets)	Encloses required options or variables.
{ } (braces)	Encloses optional options or variables.

**Note**

Do not enter the ellipsis, brackets, vertical bar, or braces in command lines. This guide uses these symbols only to show the types of entries.

CLI commands and options are in lowercase and case-sensitive. For example, when you enter the **configure** command, enter it all in lowercase, not CONFIGURE or Configure. Text entries you create are also case-sensitive. For example, if you set a username to Sys1, enter it exactly, not sys1 or SYS1.

**Note**

When Cisco Systems makes syntax changes to existing commands, the CSS automatically updates the startup-config file with the most recent changes when you upgrade the software on the CSS. For example, the CSS changes the **web-mgmt state enabled** command to the **no restrict xml** command.

Variable Argument Conventions

Some commands require variable arguments for information you must supply. CLI command variable arguments generally consist of integers, quoted and unquoted text strings, IP addresses and subnet masks, host names, Media Access Control (MAC) addresses, interfaces, stack layers, and timeslots.

[Table 1-2](#) lists the types of arguments you may encounter and the conventions to enter this information.

Table 1-2 Variable Arguments

Variable Argument	Convention
host names	Enter host names in mnemonic host-name format. For example: <code>myhost.mydomain.com</code>
integers	Enter only whole numbers with no decimal points. For example: <code>200</code>

Table 1-2 Variable Arguments (continued)

Variable Argument	Convention
Internet Protocol (IP) Addresses and Subnet Masks	<p>Enter IP addresses and subnet masks in dotted-decimal notation. This notation is four groups of up to three decimal numbers, separated by periods. Each group has a maximum number of 255. For example:</p> <pre>192.168.11.1 255.255.255.0</pre> <p>For some arguments, you can also use Classless Interdomain Routing (CIDR) notation for subnet masks. For example:</p> <ul style="list-style-type: none"> • /24 is equivalent to 255.255.255.0 • /16 is equivalent to 255.255.0.0
Interface	<p>Interface entries specify physical interfaces present in the CSS. Enter interfaces in groups of one to three characters. The format depends on your CSS.</p> <ul style="list-style-type: none"> • For the CSS 11501, CSS 11050, or CSS 11150, enter: <p><i>interface type-port</i></p> For example: <pre>e2</pre> • For the CSS 11503, CSS 11506, or CSS 11800, enter: <p><i>slot/port</i></p> For example: <pre>1/1 13/5</pre>
MAC addresses	<p>Enter MAC addresses as six groups of two hexadecimal numbers each, separated by hyphens. The alphabetic characters are not case-sensitive. For example:</p> <pre>07-77-25-C9-af-13</pre>

Table 1-2 Variable Arguments (continued)

Variable Argument	Convention
stack layers	Stack layer entries specify layers within interface stacks. Enter one of the following: <ul style="list-style-type: none"> • circuit, an CSS circuit • physical, The physical interface
text strings: quoted	Enter quoted text strings as text and spaces enclosed in quotation marks. For example: "Server lab A-3"
text strings: unquoted	Enter unquoted text strings as contiguous alphanumeric characters without spaces or quotation marks. For example: Sys_1 MyLink

Entering Multiple Commands

You can enter multiple commands on a single command line by separating them with the semicolon (;) character. For example:

```
copy running-config startup-config;archive startup-config
```

CLI Command Keyboard Shortcuts

Table 1-3 lists the CLI keyboard shortcuts to help you enter and edit command lines.

Table 1-3 CLI Command Keyboard Shortcuts

To do this...		Press these keys...
Cancel the current operation, additional display of MORE output, or delete the current line.		CTRL-c
Capitalize the character at the cursor.		ESC-c
Change:	The word at the cursor to lowercase.	ESC-l
	The word at the cursor to uppercase.	ESC-u
Delete:	A character at the cursor.	CTRL-d
	A character to the left of the cursor.	CTRL-h, Delete, or Backspace
	All characters from the cursor to the beginning of the line.	CTRL-u or CTRL-x
	All characters from the cursor to the end of the line.	CTRL-k
	All characters from the cursor to the end of the word.	ESC-d
	The word to left of the cursor.	CTRL-w or ESC-Backspace
Display the buffer's:	Next line.	CTRL-n or down-arrow
	Previous line.	CTRL-p or up-arrow

Table 1-3 CLI Command Keyboard Shortcuts (continued)

To do this...	Press these keys...	
Display MORE output:	Current line number.	l or L
	Exit from MORE output.	q, Q, or CTRL-C
	Go to a line number in the buffer. The default is the last line. To go to a specific line, enter the line number before pressing the g or G key. For example, enter 3G to go to the third line in the buffer.	g or G
	Help screen.	h or H
	Next additional screen. The default is one screen. To display more than one screen, enter a number before pressing the SPACEBAR key.	SPACEBAR
	Next line. The default is one line. To display more than one line, enter the number before pressing the RETURN key.	RETURN
	Previous line. To display more than one line, enter the number before pressing the up-arrow key.	UP-ARROW
	Previous screen. The default is one screen. To display more than one screen, enter a number before pressing the b, B, or CTRL-B key.	b, B, or CTRL-b
	Redraw screen.	r, R, or CTRL-L
	Search forward for <i>regular_expression</i> .	/ <i>regular_expression</i>
Search backward for <i>regular_expression</i> .	? <i>regular_expression</i>	
Enter a ENTER or RETURN key character.	CTRL-m	
Enter a global configuration mode <i>command</i> from any mode.	@ <i>command</i>	
Expand the command or abbreviation.	CTRL-i or TAB	

Table 1-3 CLI Command Keyboard Shortcuts (continued)

To do this...		Press these keys...
Move the cursor:	One character to the left (back).	CTRL-b or LEFT-ARROW
	One character to the right (forward).	CTRL-f or RIGHT-ARROW
	One word to the left (back), to the beginning of current or previous word.	ESC-b
	One word to the right (forward), to the end of the current or next word.	ESC-f
	To the beginning of the line.	CTRL-a
	To the end of the line.	CTRL-e
Redisplay the current line.		CTRL-l or CTRL-r
Return to the SuperUser mode from any configuration mode.		CTRL-z
Toggle:	Line logging suppression.	ESC-s
	MORE support.	ESC-m
Transpose a character at the cursor with a character to left of the cursor.		CTRL-t

Using grep

You can apply **grep** to any CLI commands with the (|) pipe character. For example:

```
show running-config|grep IP
```

```
show log log.file|grep -i ip
```

The usage of **grep** is:

```
grep [-[ilvls]] keyword
```

The options are:

- **i**, case-insensitive search
- **v**, display all lines not containing keyword
- **s**, display all lines following match

Understanding CLI Syntax Checking and Error Messages

If you enter an invalid or incomplete command, the CLI responds with a pointer (^) and error message. For example:

```
# bobo
^
%% Invalid input detected at '^' marker.
   (config)# service
           ^
%% Insufficient arguments to form command.
```

Getting CLI Help

The CLI provides several types of context-sensitive help:

- Question mark
- TAB key
- Help commands

Question Mark (?)

The question mark (?) character allows you to get the following type of help about a command at the command line.

To get help for..	Enter..
All commands for that mode	? at command prompt
All options for a command	<i>command ?</i>
All arguments for a command and its option	<i>command option ?</i>
All commands that begin with specific letters	<i>command-abbrev?</i>

If the CLI is unable to provide question mark support, a bell sounds when you enter the key.

TAB Key

When you press the TAB key or CTRL-i at the end of a unique command or option abbreviation, the CLI completes the command or options for you. For example:

```
#a1<TAB>  
#alias
```

Pressing the TAB key or CTRL-i keys also completes an option up to the point where it is unique. If multiple commands have the same abbreviation you entered, the CLI lists all of these commands.

**Note**

If the CLI is unable to provide complete TAB key support, a bell sounds when you enter the key.

Help Command

You can display a series of help topics by entering the **help** command at the CLI prompt, or display help information about specific topics including the following:

- Entering commands. Use the **help commands** command.
- Configuration files. Use the **help configuration** command.
- Keyboard shortcuts. Use the **help keys** command.
- Navigating modes. Use the **help modes** command.
- Variables. Use the **help variables** command.

User-Defined Variables

The CLI supports user-defined variables for use from the command line and from scripts. There are two types of variables: character and numeric. If you assign the variable with all integers and no spaces, it is a numeric variable. If you assign the variable with any text characters and spaces, it is a character variable.

To create or manipulate variables, refer to the **set**, **input**, **modify**, and **var-shift** commands in Chapter 2, [CLI Commands](#).

CSS Scripts

CSS scripts include scripts that you write using CLI commands, scripts provided with your CSS, and special scripts containing user profile information. For detailed information about writing scripts, refer to the *Content Services Switch Basic Configuration Guide*.

Writing and Running CLI Scripts

Use the CLI **script record** command to record command entries in a script file. You can also use an ASCII text editor to write CLI scripts (for example, Microsoft Notepad, MS-DOS Edit, UNIX PICO or EMACS). Do not use a word processing program such as Microsoft Word or WordPad.

The script commands provided through the CLI are:

- **echo**
- **endbranch**
- **exit**
- **function**
- **if**
- **input**
- **modify**
- **pause**
- **set**
- **while**

For more information about these commands and their options, refer to Chapter 2, [CLI Commands](#).

When you finished creating the file, Ctrl-C out of the **script record** command mode. If you used a text editor, save the script with any filename and extension you wish with a maximum of 32 characters. Then, use the **copy** command to move the script file to the CSS.

To run a CLI script, use the **script play** command.

CSS-Provided Scripts

The CSS contains scripts that CSS provides to assist you with, for example, CLI setup and upgrade. To see a list of CSS-provided scripts, use the **show script** command. To run a CLI script, use the **script play** command.

Aliases are also provided to run the scripts. To see a list of all aliases, refer to the **show aliases** command.

Profile Scripts

When a user logs into the CSS, the CSS runs a profile script. These scripts contain commands that are exclusive to the current CLI session. The CSS runs:

- A default-profile script for everyone
- A user-profile script for the matching username

After you log in to the CSS, you can modify your profile by changing the CLI prompt, terminal parameters, or expert mode setting, or by adding alias commands. The CSS keeps these changes in a temporary running profile until you exit the CLI.

To permanently save any running profile changes to your user profile, you can either:

- Copy the running profile to your user-profile script with the **copy running-config** command.
- Wait until you exit the CLI, and enter a **y** in response to the prompt and query to save your profile changes; if you enter an **n**, your profile changes are discarded.

For more information on changing your profile, refer to the *Content Services Switch Administration Guide*.

