



Using the Command-Line Interface

This chapter describes the Cisco IOS command-line interface (CLI) and how to use it to configure the Cisco ASR 901 router.

- [Understanding Command Modes, on page 1](#)
- [Understanding the Help System, on page 3](#)
- [Understanding Abbreviated Commands, on page 3](#)
- [Understanding no and default Forms of Commands, on page 4](#)
- [Understanding CLI Error Messages, on page 4](#)
- [Using Command History, on page 4](#)
- [Using Editing Features, on page 6](#)
- [Searching and Filtering Output of show and more Commands, on page 8](#)
- [Accessing the CLI, on page 9](#)
- [Saving Configuration Changes, on page 9](#)

Understanding Command Modes

The Cisco IOS user interface is divided into different modes. The commands depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands for each command mode.

When you start a session on the router, you begin in the user mode, often called user EXEC mode. Only a limited subset of the commands are available in user EXEC mode. For example, most of the user EXEC commands are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The user EXEC commands are not saved when the router reboots.

To gain access to all the commands, enter privileged EXEC mode. You need to enter a password to enter privileged EXEC mode. From this mode, you can enter any privileged EXEC command or enter global configuration mode.

Using the configuration modes (global, interface, and line), you can make changes to the running configuration. When you save the configuration, these commands are stored and used for router reboots. To access the various configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and line configuration mode.

[Table 1: Command Mode Summary, on page 2](#) describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode. The examples in the table use the hostname *Router*.

For more detailed information on the command modes, see the command reference guide for this release.

Table 1: Command Mode Summary

Command Mode	Access Method	Router Prompt Displayed	Exit Method	About This Mode
User EXEC	Log in.	Router>	Use the logout command.	Use this mode to: <ul style="list-style-type: none"> • Change terminal settings. • Perform basic tests. • Display system information.
Privileged EXEC	From user EXEC mode, use the enable command.	Router#	To go to user EXEC mode, use the disable , exit , or logout command.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.
Global configuration	From the privileged EXEC mode, use the configure terminal command.	Router (config)#	To go to privileged EXEC mode, use the exit or end command, or press Ctrl-Z .	Use this mode to configure parameters that apply to the entire router.
Interface configuration	From the global configuration mode, use the interface command (with a specific interface).	Router (config-if)#	To go to global configuration mode, use the exit command. To return directly to privileged EXEC mode, press Ctrl-Z .	Use this mode to configure parameters for the Ethernet ports.
VLAN configuration	While in global configuration mode, enter the vlan <i>vlan-id</i> command.	Router(config-vlan)#	To go to global configuration mode, enter the exit command. To return to privileged EXEC mode, press Ctrl-Z or use the end command.	Use this mode to configure VLAN parameters.
Line configuration	While in global configuration mode, specify a line by using the line vty or line console command.	Router(config-line)#	To go to global configuration mode, use the exit command. To return to privileged EXEC mode, press Ctrl-Z or enter end .	Use this mode to configure parameters for the terminal line.

Understanding the Help System

Enter a question mark (?) at the system prompt to display a list of commands available for each command mode. You can also obtain a list of associated keywords and arguments for any command, as shown in [Table 2: Help Summary](#), on page 3.

Table 2: Help Summary

Command	Purpose
help	Obtain a brief description of the help system in any command mode.
<i>abbreviated-command-entry?</i>	Obtain a list of commands that begin with a particular character string. For example: Router# di? dir disable disconnect
<i>abbreviated-command-entry <Tab></i>	Complete a partial command name. For example: Router# sh conf <tab> Router# show configuration
?	List all commands available for a particular command mode. For example: Router> ?
<i>command ?</i>	List the associated keywords for a command. For example: Router> show ?
<i>command keyword ?</i>	List the associated arguments for a keyword. For example: Router(config)# cdp holdtime ? <10-255> Length of time (in sec) that receiver must keep this packet

Understanding Abbreviated Commands

You need to enter only enough characters for the router to recognize the command as unique.

This example shows how to use the **show configuration** privileged EXEC command in an abbreviated form:

```
Router# show conf
```

Understanding no and default Forms of Commands

Almost every configuration command also has a **no** form. In general, use the **no** form to disable a feature or function, or reverse the action of a command. For example, the **no shutdown** interface configuration command reverses the shutdown of an interface. Use the command without the keyword **no** to re-enable a disabled feature or to enable a feature that is disabled by default.

Configuration commands can also have a **default** form. The **default** form of a command returns the command setting to its default. Most commands are disabled by default, so the **default** form is the same as the **no** form. However, some commands are enabled by default and have variables set to certain default values. In these cases, the **default** command enables the command and sets variables to their default values.

Understanding CLI Error Messages

The following table lists some error messages that you might encounter while using the CLI to configure your router.

Table 3: Common CLI Error Messages

Error Message	Meaning	How to Get Help
% Ambiguous command: "show con"	You did not enter enough characters for your router to recognize the command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
% Incomplete command.	You did not enter all the keywords or values required by this command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
% Invalid input detected at '^' marker.	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Using Command History

The software provides a history or record of commands that you entered. The command history feature is particularly useful for recalling long or complex commands or entries, including access lists. You can customize this feature to suit your needs as described in these sections:

Changing the Command History Buffer Size

By default, the router records ten command lines in its history buffer. You can alter this number for a current terminal session or for all sessions on a particular line. These procedures are optional.

Beginning in privileged EXEC mode, enter this command to change the number of command lines that the router records during the current terminal session:

```
Router# terminal history
size
    number-of-lines
```

The range is from 0 to 256.

Beginning in line configuration mode, enter this command to configure the number of command lines the router records for all sessions on a particular line:

```
Router(config-line)# history
[
size
    number-of-lines
```

The range is from 0 to 256.

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in [Table 4: Recalling Commands](#), on page 5. These actions are optional.

Table 4: Recalling Commands

Action ¹	Result
Press Ctrl-P or the up arrow key.	Recall commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Press Ctrl-N or the down arrow key.	Return to more recent commands in the history buffer after recalling commands with Ctrl-P or the up arrow key. Repeat the key sequence to recall successively more recent commands.
show history	While in privileged EXEC mode, list the last several commands that you just entered. The number of commands that appear is controlled by the setting of the terminal history global configuration command and the history line configuration command.

¹ The arrow keys function only on ANSI-compatible terminals such as VT100s.

Disabling the Command History Feature

The command history feature is automatically enabled. You can disable it for the current terminal session or for the command line. These procedures are optional.

To disable the feature during the current terminal session, use the **terminal no history** privileged EXEC command.

To disable command history for the line, use the **no history** line configuration command.

Using Editing Features

This section contains the following the editing features that can help you manipulate the command line.

Enabling and Disabling Editing Features

Although the enhanced editing mode is automatically enabled, you can disable it, re-enable it, or configure a specific line to have enhanced editing. These procedures are optional.

To globally disable enhanced editing mode, enter this command in line configuration mode:

```
Router (config-line)# no editing
```

To re-enable the enhanced editing mode for the current terminal session, enter this command in privileged EXEC mode:

```
Router# terminal editing
```

To reconfigure a specific line to have enhanced editing mode, enter this command in line configuration mode:

```
Router (config-line)# editing
```

Editing Commands through Keystrokes

[Table 5: Editing Commands through Keystrokes](#), on page 6 shows the keystrokes that you need to edit command lines. These keystrokes are optional.

Table 5: Editing Commands through Keystrokes

Capability	Keystroke ²	Purpose
Move around the command line to make changes or corrections.	Press Ctrl-B , or press the left arrow key.	Move the cursor back one character.
	Press Ctrl-F , or press the right arrow key.	Move the cursor forward one character.
	Press Ctrl-A .	Move the cursor to the beginning of the command line.
	Press Ctrl-E .	Move the cursor to the end of the command line.
	Press Esc B .	Move the cursor back one word.
	Press Esc F .	Move the cursor forward one word.
	Press Ctrl-T .	Transpose the character to the left of the cursor with the character located at the cursor.

Capability	Keystroke ²	Purpose
Recall commands from the buffer and paste them in the command line. The router provides a buffer with the last ten items that you deleted.	Press Ctrl-Y .	Recall the most recent entry in the buffer.
	Press Esc Y .	Recall the next buffer entry. The buffer contains only the last 10 items that you have deleted or cut. If you press Esc Y more than ten times, you cycle to the first buffer entry.
Delete entries if you make a mistake or change your mind.	Press the Delete or Backspace key.	Erase the character to the left of the cursor.
	Press Ctrl-D .	Delete the character at the cursor.
	Press Ctrl-K .	Delete all characters from the cursor to the end of the command line.
	Press Ctrl-U or Ctrl-X .	Delete all characters from the cursor to the beginning of the command line.
	Press Ctrl-W .	Delete the word to the left of the cursor.
	Press Esc D .	Delete from the cursor to the end of the word.
Capitalize or lower the case or capitalize a set of letters.	Press Esc C .	Capitalize at the cursor.
	Press Esc L .	Change the word at the cursor to lowercase.
	Press Esc U .	Capitalize letters from the cursor to the end of the word.
Designate a particular keystroke as an executable command, perhaps as a shortcut.	Press Ctrl-V or Esc Q .	
Scroll down a line or screen on displays that are longer than the terminal screen can display. Note The More prompt is used for any output that has more lines than can be displayed on the terminal screen, including show command output. You can use the Return and Space bar keystrokes whenever you see the More prompt.	Press the Return key.	Scroll down one line.
	Press the Space bar.	Scroll down one screen.

Capability	Keystroke ²	Purpose
Redisplay the current command line if the router suddenly sends a message to your screen.	Press Ctrl-L or Ctrl-R .	Redisplay the current command line.

² The arrow keys function only on ANSI-compatible terminals such as VT100s.

Editing Command Lines that Wrap

You can use a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command. The keystroke actions are optional.

To scroll back to the beginning of the command entry, press **Ctrl-B** or the left arrow key repeatedly. You can also press **Ctrl-A** to immediately move to the beginning of the line.



Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

In this example, the **access-list** global configuration command entry extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign (\$) shows that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.

```
Router(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1
Router(config)# $ 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.25
Router(config)# $t tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq
Router(config)# $108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq 45
```

After you complete the entry, press **Ctrl-A** to check the complete syntax before pressing the **Return** key to execute the command. The dollar sign (\$) appears at the end of the line to show that the line has been scrolled to the right:

```
Router(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1$
```

The software assumes you have a terminal screen that is 80 columns wide. If you have a width other than that, use the **terminal width** privileged EXEC command to set the width of your terminal.

Use line wrapping with the command history feature to recall and modify previous complex command entries. For information about recalling previous command entries, see the [Editing Commands through Keystrokes, on page 6](#).

Searching and Filtering Output of show and more Commands

You can search and filter the output for **show** and **more** commands. This is useful when you need to sort through large amounts of output or if you want to exclude output that you do not need to see. Using these commands is optional.

To use this functionality, use **show** or **more** command followed by the *pipe* character (`|`), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search for or filter out:

```
command| {begin | include | exclude} regular-expression
```

Expressions are case sensitive. For example, if you use **exclude output** command, the lines that contain *output* are not displayed, but the lines that contain *Output* appear.

This example shows how to include in the output display only lines where the expression *protocol* appears:

```
Router# show interfaces | include protocol
Vlan1 is up, line protocol is up
Vlan10 is up, line protocol is down
GigabitEthernet0/1 is up, line protocol is down
GigabitEthernet0/2 is up, line protocol is up
```

Accessing the CLI

You can access the CLI through a console connection, through Telnet, or by using the browser.

Accessing the CLI through a Console Connection or through Telnet

Before accessing the CLI, you must connect a terminal or PC to the router console port and power on the router as described in the hardware installation guide that shipped with your router.

If your router is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your router must first be configured for this type of access..

You can use one of these methods to establish a connection with the router:

- Connect the router console port to a management station or dial-up modem. For information about connecting to the console port, see the router hardware installation guide.
- Use any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The router must have network connectivity with the Telnet or SSH client, and the router must have an enable secret password configured.

The router supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.

The router supports up to five simultaneous secure SSH sessions.

After you connect through the console port, through a Telnet session or through an SSH session, the user EXEC prompt appears on the management station.

Saving Configuration Changes

To save your configuration changes to NVRAM, so that the changes are not lost during a system reload or power outage, enter the **copy running-config startup-config** command. For example:

```
Router# copy running-config startup-config
Router# write memory
Building configuration...
```

It might take a few minutes to save the configuration to NVRAM. After the configuration has been saved, the following message appears:

```
[OK]  
Router#
```

For additional information about using the Cisco IOS Release 15.1SNG, see the guides listed at:

http://www.cisco.com/en/US/products/ps11280/tsd_products_support_series_home.html