



## Basic Command-Line Interface Commands

---

This chapter describes the commands used to enter and exit the various Cisco IOS configuration command modes. It provides a description of help features, command-line interface (CLI) navigation commands, and the command history feature.

The CLI allows you to enter partial Cisco IOS configuration commands. The software recognizes a command when you enter enough characters of the command to uniquely identify it.

For user interface task information and examples, see the “Using the Command-Line Interface” chapter of the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2*.

# do

To execute an EXEC-level command from global configuration mode or any configuration submode, use the **do** command in any configuration mode.

**do** *command*

<b>Syntax Description</b>	<i>command</i>	The EXEC command to be executed.
---------------------------	----------------	----------------------------------

<b>Defaults</b>	No default behavior or values.
-----------------	--------------------------------

<b>Command Modes</b>	All Configuration Modes
----------------------	-------------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.2(8)T	This command was introduced.

<b>Usage Guidelines</b>	Use this command to execute EXEC commands (such as <b>show</b> , <b>clear</b> , and <b>debug</b> commands) while configuring your routing device. After the EXEC command is executed, the system will return to the configuration mode you were using.
-------------------------	--



**Tip**

This command can be useful for saving your configuration to the startup-config file without having to return to the EXEC mode (**do copy running-config startup-config**), or for checking the status of a feature (using a **do show** command) while configuring the feature.

## Examples

In the following example, the **show interfaces** EXEC command is issued from within global configuration mode:

```
Router(config)# do show interfaces serial 3/0

Serial3/0 is up, line protocol is up
  Hardware is M8T-RS232
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input never, output 1d17h, output hang never
  Last clearing of "show interface" counters never
  .
  .
  .
Router(config)#
```

In the following example, the **clear vpdn tunnel** EXEC command is issued from within VPDN configuration mode:

```
Router(config-vpdn)# do clear vpdn tunnel
Router(config-vpdn)#
```

# disable

To exit privileged EXEC mode and return to user EXEC mode, or to exit to a lower privilege level, enter the **disable** command in EXEC mode.

**disable** [*privilege-level*]

<b>Syntax Description</b>	<i>privilege-level</i> (Optional) Specific privilege level (other than user EXEC mode).
---------------------------	---

<b>Command Modes</b>	EXEC
----------------------	------

<b>Command History</b>	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** Up to 16 security levels can be configured using Cisco IOS software. If such levels are configured on a system, using this command with the *privilege-level* option allows you to exit to a lower security level. If a level is not specified, the user will exit to the user EXEC mode, which is the default.



**Note**

Five EXEC commands are associated with privilege level 0: **disable**, **enable**, **exit**, **help**, and **logout**. If you configure a privilege level greater than 0, these five commands will not be included in the command set for that privilege level.

**Examples** In the following example, the user enters privileged EXEC mode using the **enable** command, then exits back to user EXEC mode using the **disable** command. Note that the prompt for user EXEC mode is >, and the prompt for privileged EXEC mode is #.

```
Router> enable
Password: <letmein>
Router# disable
Router>
```

<b>Related Commands</b>	Command	Description
	<a href="#">enable</a>	Enables higher privilege level access, such as privileged EXEC mode.

# editing

To reenable Cisco IOS enhanced editing features for a particular line after they have been disabled, use the **editing** command in line configuration mode. To disable these features, use the **no** form of this command.

**editing**

**no editing**

---

**Syntax Description** This command has no arguments or keywords.

---

**Defaults** Enabled

---

**Command Modes** Line configuration

---

Release	Modification
10.0	This command was introduced.

---



---

**Usage Guidelines** Enhanced editing features are enabled by default. However, there may be situations in which you need to disable these features. The **no** form of this command disables these enhanced editing features, and the plain form of the command can be used to reenable these features.

[Table 1](#) provides a description of the keys used to enter and edit commands when the editing features are enabled. Ctrl indicates the Control key, which must be pressed simultaneously with its associated letter key. Esc indicates the Escape key, which must be pressed first, followed by its associated letter key. A comma is used in the following table to indicate a key sequence (the comma key should not be pressed). Keys are not case sensitive. Many letters used for CLI navigation and editing were chosen to provide an easy way of remembering their functions. In the following table ([Table 1](#)), characters are bolded in the “Function Summary” column to indicate the relation between the letter used and the function.

**Table 1** *Command Editing Keys and Functions*

Keys	Function Summary	Function Details
Tab	Complete command	Completes a partial command name entry. When you enter a unique set of characters and press the Tab key, the system completes the command name. If you enter a set of characters that could indicate more than one command, the system beeps to indicate an error. To view the commands which match the set of characters you have entered, enter a question mark (?) immediately following the partial command (no space). The CLI will then list the commands that begin with that string.
Return (at the command line)	Execute	Executes the command.
Return (at the --More-- prompt)	Continue	Displays the next line of output.
Space Bar (at the --More-- prompt)	Continue	Displays the next screen of output. The amount of output you see will depend on the screen depth setting of your terminal.
Delete or Backspace	Backspace	Erases the character to the left of the cursor.
Left Arrow <sup>1</sup> or Ctrl-B	Back character	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.
Right Arrow <sup>1</sup> or Ctrl-F	Forward character	Moves the cursor one character to the right.
Esc, B	Back word	Moves the cursor back one word.
Esc, F	Forward word	Moves the cursor forward one word.
Ctrl-A	Beginning of line	Moves the cursor to the beginning of the line.
Ctrl-E	End of line	Moves the cursor to the end of the command line.
Ctrl-D	Delete character	Deletes the character at the cursor.
Esc, D	Delete next word	Deletes from the cursor to the end of the word .
Ctrl-W	Delete previous word	Deletes the word to the left of the cursor.
Ctrl-K	Delete line forward	Deletes all characters from the cursor to the end of the command line.
Ctrl-U or Ctrl-X	Delete line backward	Deletes all characters from the cursor back to the beginning of the command line.
Ctrl-T	Transpose characters	Transposes the character to the left of the cursor with the character located at the cursor.

**Table 1** Command Editing Keys and Functions (continued)

Keys	Function Summary	Function Details
Ctrl-R or Ctrl-L	Redisplay line	Redisplays the system prompt and command line.
Ctrl-V or Esc, Q	Ignore editing	Inserts a code to indicate to the system that the keystroke immediately following should be treated as a command entry, <i>not</i> as an editing key.
Up Arrow <sup>1</sup> or Ctrl-P	Previous command	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down Arrow <sup>1</sup> or Ctrl-N (next)	Next command	Returns to more recent commands in the history buffer (after recalling commands with the Up Arrow or Ctrl-P). Repeat the key sequence to recall successively more recent commands.
Ctrl-Y	Recall last deleted command	Recalls the most recent entry in the delete buffer. The delete buffer contains the last ten items you have deleted or cut. Ctrl-Y can be used in conjunction with Esc Y.
Esc, Y	Recall next deleted command	Recalls the next entry in the delete buffer. The delete buffer contains the last ten items you have deleted. Press Ctrl-Y first to recall the most recent entry. Then press Esc Y up to nine times to recall the remaining entries in the buffer. If you bypass an entry, continue to press Esc Y to cycle back to it.
Esc, C	Capitalize word	Capitalizes the word from the cursor to the end of the word.
Esc, U	Make word uppercase	Changes all letters from the cursor to the next space on the line appear in uppercase letters.
Esc, L	Make word lowercase	Changes the word to lowercase from the cursor to the end of the word.

1. The arrow keys function only with ANSI-compatible terminals.

### Examples

In the following example, enhanced editing mode is disabled on line 3:

```
Router(config)# line 3
Router(config-line)# no editing
```

### Related Commands

Command	Description
<a href="#">terminal editing</a>	Controls CLI enhanced editing feature for the current terminal session.

# enable

To enter privileged EXEC mode, or any other security level set by a system administrator, use the **enable** command in EXEC mode.

**enable** [*privilege-level*]

<b>Syntax Description</b>	<i>privilege-level</i> (Optional) Privilege level at which to log in.				
<b>Defaults</b>	Privilege-level 15 (privileged EXEC)				
<b>Command Modes</b>	EXEC				
<b>Command History</b>	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>10.0</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	10.0	This command was introduced.
Release	Modification				
10.0	This command was introduced.				

## Usage Guidelines

Entering privileged EXEC mode enables the use of privileged commands. Because many of the privileged commands set operating parameters, privileged access should be password-protected to prevent unauthorized use. If the system administrator has set a password with the **enable password** global configuration command, you are prompted to enter it before being allowed access to privileged EXEC mode. The password is case sensitive.

If an **enable** password has not been set, enable mode only can be accessed through the console connection.

Security levels can be set by an administrator using the **enable password** and **privilege level** commands. Up to 16 privilege levels can be specified, using the numbers 0 through 15. Using these privilege levels, the administrator can allow or deny access to specific commands. Privilege level 0 is associated with user EXEC mode, and privilege level 15 is associated with privileged EXEC mode.

For more information on defined privilege levels, see the “Passwords and Privileges” chapters of the *Cisco IOS Security Configuration Guide* and the *Cisco IOS Security Command Reference* publications.

If a level is not specified when entering the **enable** command, the user will enter the default mode of privileged EXEC (level 15).

**Examples**

In the following example, the user enters privileged EXEC mode using the **enable** command. The system prompts the user for a password before allowing access to the privileged EXEC mode. The password is not printed to the screen. The user then exits back to user EXEC mode using the **disable** command. Note that the prompt for user EXEC mode is >, and the prompt for privileged EXEC mode is #.

```
Router> enable
Password: <letmein>
Router# disable
Router>
```

**Related Commands**

Command	Description
<b>disable</b>	Exits from privileged EXEC mode to user EXEC mode, or, if privilege levels are set, to the specified privilege level.
<b>enable password</b>	Sets a local password to control access to various privilege levels.
<b>privilege level (global)</b>	Sets a privilege level for a command.
<b>privilege level (line)</b>	Sets a privilege level for a command for a specific line.

# end

To end the current configuration session and return to privileged EXEC mode, use the **end** command in global configuration mode.

**end**

**Syntax Description** This command has no arguments or keywords.

**Defaults** No default behavior or values

**Command Modes** Global configuration

Command History	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** This command will bring you back to privileged EXEC mode regardless of what configuration mode or configuration submode you are in.



**Note**

This global configuration command can be used in any configuration mode.

Use this command when you are done configuring the system and you want to return to EXEC mode to perform verification steps.

**Examples** In the following example, the **end** command is used to exit from ALPS ASCU configuration mode and return to privileged EXEC mode. A **show** command is used in privileged EXEC mode to verify the configuration.

```
Router# configure terminal
Router(config)# interface serial 1:1
Router(config-if)# alps ascu 4B
Router(config-alps-ascu)# end
Router# show interface serial 1:1
```

Related Commands	Command	Description
	<b>exit (global)</b>	Exits from the current configuration mode.

## exit (EXEC)

To close an active terminal session by logging off the router, use the **exit** command in EXEC mode.

**exit**

---

**Syntax Description** This command has no arguments or keywords.

---

**Defaults** No default behavior or values

---

**Command Modes** EXEC

---

Command History	Release	Modification
	10.0	This command was introduced.

---



---

**Usage Guidelines** Use the **exit** (EXEC) command in EXEC mode to exit the active session (log off the device).

---

**Examples** In the following example, the **exit** (global) command is used to move from global configuration mode to privileged EXEC mode, the **disable** command is used to move from privileged EXEC mode to user EXEC mode, and the **exit** (EXEC) command is used to log off (exit the active session):

```
Router(config)# exit
Router# disable
Router> exit
```

---

Related Commands	Command	Description
	<b>disconnect</b>	Disconnects a line.
	<b>end</b>	Exits configuration mode, or any of the configuration submodes.
	<b>exit (global)</b>	Exits from the current configuration mode to the next highest configuration mode.

---

# exit (global)

To exit any configuration mode to the next highest mode in the CLI mode hierarchy, use the **exit** command in any configuration mode.

**exit**

**Syntax Description** This command has no arguments or keywords.

**Defaults** No default behavior or values

**Command Modes** All configuration modes

Command History	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** The **exit** command is used in the Cisco IOS CLI to exit from the current command mode to the next highest command mode in the CLI mode hierarchy.

For example, use the exit command in global configuration mode to return to privileged EXEC mode. Use the exit command in interface, line, or router configuration mode to return to global configuration mode. Use the exit command in subinterface configuration mode to return to interface configuration mode. At the highest level, EXEC mode, the exit command will exit the EXEC mode and disconnect from the router interface (see the description of the **exit (EXEC)** command for details).

**Examples** The following example displays an exit from the subinterface configuration mode to return to the interface configuration mode:

```
Router(config-subif)# exit
Router(config-if)#
```

The following example displays an exit from the interface configuration mode to return to the global configuration mode:

```
Router(config-if)# exit
Router(config)#
```

Related Commands	Command	Description
	<b>disconnect</b>	Disconnects a line.
	<b>end</b>	Exits from any configuration mode to privileged EXEC mode.
	<b>exit (EXEC)</b>	Closes the active terminal session by logging off the router.

# full-help

To get help for the full set of user-level commands, use the **full-help** command in line configuration mode.

## full-help

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Line configuration

Command History	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** The **full-help** command enables (or disables) an unprivileged user to see all of the help messages available. It is used with the **show ?** command.

**Examples** In the following example, the **show ?** command is used first with full-help disabled. Then full-help is enabled for the line, and the **show ?** command is used again to demonstrate the additional help output that is displayed.

```
Router> show ?

bootflash  Boot Flash information
calendar   Display the hardware calendar
clock      Display the system clock
context    Show context information
dialer     Dialer parameters and statistics
history    Display the session command history
hosts      IP domain-name, lookup style, nameservers, and host table
isdn       ISDN information
kerberos   Show Kerberos Values
modemcap   Show Modem Capabilities database
ppp        PPP parameters and statistics
rmon       rmon statistics
sessions   Information about Telnet connections
snmp       snmp statistics
terminal   Display terminal configuration parameters
users      Display information about terminal lines
version    System hardware and software status

Router> enable
Password:<letmein>

Router# configure terminal
```

```

Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# line console 0
Router(config-line)# full-help
Router(config-line)# ^Z
Router#
%SYS-5-CONFIG I: Configured from console by console
Router# disable
Router> show ?

  access-expression  List access expression
  access-lists       List access lists
  aliases            Display alias commands
  apollo             Apollo network information
  appletalk          AppleTalk information
  arp                ARP table
  async              Information on terminal lines used as router interfaces
  bootflash          Boot Flash information
  bridge             Bridge Forwarding/Filtering Database [verbose]
  bsc                BSC interface information
  bstun              BSTUN interface information
  buffers            Buffer pool statistics
  calendar           Display the hardware calendar
  .
  .
  .
  translate          Protocol translation information
  ttycap             Terminal capability tables
  users              Display information about terminal lines
  version            System hardware and software status
  vines              VINES information
  vlans              Virtual LANs Information
  whoami             Info on current tty line
  x25                X.25 information
  xns                XNS information
  xremote            XRemote statistics

```

---

**Related Commands**

Command	Description
<a href="#">help</a>	Displays a brief description of the help system.

---

# help

To display a brief description of the help system, enter the **help** command in any configuration mode.

**help**

---

**Syntax Description** This command has no arguments or keywords.

---

**Defaults** No default behavior or values

---

**Command Modes** All command modes

---

Command History	Release	Modification
	10.0	This command was introduced.

---



---

**Usage Guidelines** The **help** command provides a brief description of the context-sensitive help system, which functions as follows:

- To list all commands available for a particular command mode, enter a question mark (?) at the system prompt.
- 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 *word help*, because it lists only the keywords or arguments that begin with the abbreviation you entered.
- To list the keywords and arguments associated with a command, enter a question mark (?) in place of a keyword or argument on the command line. This form of help is called *command syntax help*, because it lists the keywords or arguments that apply based on the command, keywords, and arguments you have already entered.

---

**Examples** In the following example, the **help** command is used to display a brief description of the help system:

```
Router# help
```

```
Help may be requested at any point in a command by entering
a question mark '?'. If nothing matches, the help list will
be empty and you must backup until entering a '?' shows the
available options.
```

```
Two styles of help are provided:
```

1. Full help is available when you are ready to enter a command argument (e.g. 'show ?') and describes each possible argument.
2. Partial help is provided when an abbreviated argument is entered and you want to know what arguments match the input (e.g. 'show pr?').

The following example shows how to use word help to display all the privileged EXEC commands that begin with the letters “co”. The letters entered before the question mark are reprinted on the next command line to allow the user to continue entering the command.

```
Router# co?
configure connect copy
Router# co
```

The following example shows how to use command syntax help to display the next argument of a partially complete **access-list** command. One option is to add a wildcard mask. The <cr> symbol indicates that the other option is to press Return to execute the command without adding any more keywords or arguments. The characters entered before the question mark are reprinted on the next command line to allow the user to continue entering the command or to execute that command as it is.

```
Router(config)# access-list 99 deny 131.108.134.234 ?
  A.B.C.D Mask of bits to ignore
  <cr>
Router(config)# access-list 99 deny 131.108.134.234
```

---

**Related Commands**

Command	Description
<a href="#">full-help</a>	Gets help for the full set of user-level commands.

---

# history

To enable the command history function, use the **history** command in line configuration mode. To disable the command history feature, use the **no** form of this command.

**history**

**no history**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Enabled, ten command lines in buffer

**Command Modes** Line configuration

Command History	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** The command history feature provides a record of EXEC commands that you have entered. This feature is particularly useful for recalling long or complex commands or entries, including access lists.

To change the number of command lines that the system will record in its history buffer, use the **history size** line configuration command.

The **history** command enables the history function with the last buffer size specified or, if there was not a prior setting, with the default of ten lines. The **no history** command disables the history feature.

The **show history** EXEC command will list the commands you have entered, but you can also use your keyboard to display individual commands. [Table 2](#) lists the keys you can use to recall commands from the command history buffer.

**Table 2 History Keys**

Key(s)	Functions
Ctrl-P or Up Arrow <sup>1</sup>	Recalls commands in the history buffer in a backward sequence, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or Down Arrow <sup>1</sup>	Returns to more recent commands in the history buffer after recalling commands with Ctrl-P or the Up Arrow. Repeat the key sequence to recall successively more recent commands.

1. The arrow keys function only with ANSI-compatible terminals.

---

**Examples**

In the following example, line 4 is configured with a history buffer size of 35 lines:

```
Router(config)# line 4
Router(config-line)# history size 35
```

---

**Related Commands**

Command	Description
<a href="#">history size</a>	Sets the command history buffer size for a particular line.
<a href="#">show history</a>	Lists the commands you have entered in the current EXEC session.
<a href="#">terminal history</a>	Enables the command history feature for the current terminal session or changes the size of the command history buffer for the current terminal session.

# history size

To change the command history buffer size for a particular line, use the **history size** command in line configuration mode. To reset the command history buffer size to ten lines, use the **no** form of this command.

**history size** *number-of-lines*

**no history size**

<b>Syntax Description</b>	<i>number-of-lines</i>	Specifies the number of command lines that the system will record in its history buffer. The range is from 0 to 256. The default is ten.
---------------------------	------------------------	--

<b>Defaults</b>	Ten command lines
-----------------	-------------------

<b>Command Modes</b>	Line configuration
----------------------	--------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	10.0	This command was introduced.

**Usage Guidelines** The **history size** command should be used in conjunction with the **history** and **show history** commands. The **history** command enables or disables the command history function. The **show history** command lists the commands you have entered in the current EXEC session. The number of commands that the history buffer will show is set by the **history size** command.



**Note**

The **history size** command only sets the size of the buffer; it does not reenables the history feature. If the **no history** command is used, the **history** command must be used to reenables this feature.

**Examples** The following example displays line 4 configured with a history buffer size of 35 lines:

```
Router(config)# line 4
Router(config-line)# history size 35
```

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>history</b>	Enables or disables the command history function.
	<b>show history</b>	Lists the commands you have entered in the current EXEC session.
	<b>terminal history size</b>	Enables the command history function for the current terminal session or changes the size of the command history buffer for the current terminal session.

# menu (EXEC)

To display a preconfigured user menu, use the **menu** command in user or privileged EXEC mode.

**menu** *menu-name*

<b>Syntax Description</b>	<i>menu-name</i>	The name of the menu.
---------------------------	------------------	-----------------------

<b>Command Modes</b>	EXEC
----------------------	------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	10.0	This command was introduced.

**Usage Guidelines** A user menu is a type of user interface where text descriptions of actions to be performed are displayed to the user. The user can use the menu to select services and functions without having to know the details of command-line interface (CLI) commands.

Menus can be created for users in global configuration mode, using the commands listed in the “Related Commands” section. The description of these commands can be found in the [“Connection, Menu, and System Banner Commands”](#) chapter of this document.

A menu can be invoked at either the user or privileged EXEC level, but if an item in the menu contains a privileged EXEC command, the user must be logged in at the privileged level for the command to succeed.

**Examples** The following example invokes a menu named OnRamp:

```
Router> menu OnRamp

Welcome to OnRamp Internet Services

Type a number to select an option;
Type 9 to exit the menu.

1 Read email
2 UNIX Internet access
3 Resume UNIX connection

6 Resume next connection

9 Exit menu system
```

Related Commands	Command	Description
	<b>menu clear-screen</b>	Clears the terminal screen before displaying a menu.
	<b>menu command</b>	Specifies underlying commands for user interface menus.
	<b>menu default</b>	Specifies the menu item to use as the default.
	<b>menu line-mode</b>	Requires the user to press Enter after specifying an option number.
	<b>menu options</b>	Sets options for items in user interface menus.
	<b>menu prompt</b>	Specifies the prompt for a user interface menu.
	<b>menu single-space</b>	Displays menu items single-spaced rather than double-spaced.
	<b>menu status-line</b>	Displays a line of status information about the current user at the top of a menu
	<b>menu text</b>	Specifies the text of a menu item in a user interface menu.
	<b>menu title</b>	Creates a title, or banner, for a user menu.
	<b>no menu</b>	Deletes a specified menu from a menu configuration.

## more <url> begin

To search the output of any **more** command, use the **more URL | begin** command in EXEC mode. This command begins unfiltered output of the **more** command with the first line that contains the regular expression you specify.

**more** *URL* | **begin** *regular-expression*

Syntax Description		
<i>URL</i>		The Universal Resource Locator (url) of the file to display. <b>More</b> commands are advanced <b>show</b> commands; for details, see the command reference page in this book for the <b>more</b> command.
		A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
<i>regular-expression</i>		Any regular expression found in <b>more</b> command output.
/		Specifies a search at a --More-- prompt that begins unfiltered output with the first line that contains the regular expression.
-		Specifies a filter at a --More-- prompt that only displays output lines that do not contain the regular expression.
+		Specifies a filter at a --More-- prompt that only displays output lines that contain the regular expression.

**Command Modes** EXEC

Command History	Release	Modification
	11.3 AA	The <b>more</b> command was introduced.
	12.0(1)T	This extension of the <b>more</b> command was introduced.

**Usage Guidelines** The *regular-expression* argument is case sensitive and allows for complex matching requirements.

You can specify a new search at every --More-- prompt.

To search the remaining output of the **more** command, use the following command at the --More-- prompt:

*/regular-expression*

To filter the remaining output of the **more** command, use one of the following commands at the --More-- prompt:

*-regular-expression*

*+regular-expression*

When output volume is large, the search can produce long lists of output. To interrupt the output, press **Ctrl-^** (Ctrl-Shift-6) or **Ctrl-Z**.

**Note**

Once you specify a filter for a **more** command, you cannot specify another filter at a --More-- prompt. The first specified filter remains until the **more** command output finishes or until you interrupt the output. The use of the keyword **begin** does not constitute a filter.

Because prior output is not saved, you cannot search or filter backward through prior output.

**Examples**

The following is partial sample output of the **more nvram:startup-config | begin ip** command that begins unfiltered output with the first line that contain the regular expression “ip.” At the --More-- prompt, the user specifies a filter to exclude output lines that contain the regular expression “ip.”

```
router# more nvram:startup-config | begin ip

ip subnet-zero
ip domain-name cisco.com
ip name-server 198.92.30.32
ip name-server 171.69.2.132
!
isdn switch-type primary-5ess
.
.
.
interface Ethernet1
 ip address 5.5.5.99 255.255.255.0
--More--
-ip
filtering...
 media-type 10BaseT
!
interface Serial0:23
 encapsulation frame-relay
 no keepalive
 dialer string 4001
 dialer-group 1
 isdn switch-type primary-5ess
 no fair-queue
```

**Related Commands**

Command	Description
<b>more &lt;url&gt; exclude</b>	Filters <b>more</b> command output so that it excludes lines that contain a particular regular expression.
<b>more &lt;url&gt; include</b>	Filters <b>more</b> command output so that it displays only lines that contain a particular regular expression.
<b>show &lt;command&gt; begin</b>	Searches the output of any <b>show</b> command and displays the output from the first instance of a specified string.
<b>show &lt;command&gt; exclude</b>	Filters <b>show</b> command output so that it excludes lines that contain a particular regular expression.
<b>show &lt;command&gt; include</b>	Filters <b>show</b> command output so that it displays only lines that contain a particular regular expression.

## more <url> exclude

To filter **more** command output so that it excludes lines that contain a particular regular expression, use the **more exclude** command in EXEC mode.

**more** *file-url* | **exclude** *regular-expression*

Syntax Description	URL
	<p>The Universal Resource Locator (URL) of the file to display. <b>More</b> commands are advanced <b>show</b> commands; for details, see the command reference page in this book for the <b>more</b> command.</p> <p>The Cisco IOS File System (IFS) uses URLs to specify the location of a file system, directory, and file. Typical URL elements include:</p> <p><i>prefix:[directory]/filename</i></p> <p>Prefixes can be local file systems or file locations, such as <b>nvr</b> or <b>system</b>. Alternatively, you can specify network locations using the following syntax:</p> <p><b>ftp</b>:<i>[[/[username[:password]@]location]/directory]/filename</i></p> <p><b>tftp</b>:<i>[[/location]/directory]/filename</i></p> <p><b>rcp</b>:<i>[[/[username@]location]/directory]/filename</i></p>
	<p> </p> <p>A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.</p>
	<p><i>regular-expression</i></p> <p>Any regular expression found in <b>more</b> command output.</p>
	<p>/</p> <p>Specifies a search at a --More-- prompt that begins unfiltered output with the first line that contains the regular expression.</p>

**Command Modes** EXEC

Command History	Release	Modification
	11.3 AA	The <b>more</b> command was introduced.
	12.0(1)T	This extension of the <b>more</b> command was introduced.

**Usage Guidelines** The *regular-expression* argument is case sensitive and allows for complex matching requirements. You can specify a new search at any --More-- prompt. To search the remaining output of the **more** command, use the following command at the --More-- prompt:

*/regular-expression*

When output volume is large, the search can produce long lists of output. To interrupt the output, press **Ctrl-^** (Ctrl-Shift-6) or **Ctrl-Z**.

Because prior output is not saved, you cannot search or filter backward through prior output.

## Examples

The following is partial sample output of the **more nvram:startup-config | exclude service** command. The use of **| exclude service** in the command specifies a filter that excludes lines that contain the regular expression “service.” At the **--More--** prompt, the user searches for the regular expression “Dialer1,” which continues filtered output with the first line that contains “Dialer1.”

```
router# more nvram:startup-config | exclude service
!
version 12.0
!
hostname router
!
boot system flash
no logging buffered
!
ip subnet-zero
ip domain-name cisco.com
.
.
.
--More--
/Dialer1
filtering...
interface Dialer1
 no ip address
 no ip directed-broadcast
 dialer in-band
 no cdp enable
```

## Related Commands

Command	Description
<b>more begin</b>	Begins unfiltered output of the <b>more</b> command with the first line that contains the regular expression you specify.
<b>more include</b>	Filters <b>more</b> command output so that it displays only lines that contain a particular regular expression.
<b>show begin</b>	Searches the output of any show command and displays the output from the first instance of a specified string.
<b>show exclude</b>	Filters <b>show</b> command output so that it excludes lines that contain a particular regular expression.
<b>show include</b>	Filters <b>show</b> command output so that it displays only lines that contain a particular regular expression.

# more <url> include

To filter **more** command output so that it displays only lines that contain a particular regular expression, use the **more include** command in EXEC mode.

**more** *file-url* | **include** *regular-expression*

Syntax Description	<i>file-url</i>	The Universal Resource Locator (url) of the file to display. <b>More</b> commands are advanced <b>show</b> commands; for details, see the command reference page in this book for the <b>more</b> command.
		A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
	<i>regular-expression</i>	Any regular expression found in <b>more</b> command output.
	/	Specifies a search at a --More-- prompt that begins unfiltered output with the first line that contains the regular expression.

**Command Modes** EXEC

Command History	Release	Modification
	11.3 AA	The <b>more</b> command was introduced.
	12.0(1)T	This extension of the <b>more</b> command was introduced.

**Usage Guidelines** The *regular-expression* argument is case sensitive and allows for complex matching requirements. You can specify a new search at any --More-- prompt. To search the remaining output of the **more** command, use the following syntax at the --More-- prompt:

*/regular-expression*

When output volume is large, the search can produce long lists of output. To interrupt the output, press **Ctrl-^** (Ctrl-Shift-6) or **Ctrl-Z**.

Because prior output is not saved, you cannot search or filter backward through prior output.

**Examples** The following is partial sample output of the **more nvram:startup-config | include ip** command. It only displays lines that contain the regular expression “ip.”

```
router# more nvram:startup-config | include ip
ip subnet-zero
ip domain-name cisco.com
ip name-server 198.92.30.32
ip name-server 171.69.2.132
description ip address 172.21.53.199 255.255.255.0
ip address 172.21.53.199 255.255.255.0
```

Related Commands	Command	Description
	<b>more &lt;url&gt; begin</b>	Begins unfiltered output of the <b>more</b> command with the first line that contains the regular expression you specify.
	<b>more &lt;url&gt; exclude</b>	Filters <b>more</b> command output so that it excludes lines that contain a particular regular expression.
	<b>show &lt;command&gt; begin</b>	Searches the output of any show command and displays the output from the first instance of a specified string.
	<b>show &lt;command&gt; exclude</b>	Filters <b>show</b> command output so that it excludes lines that contain a particular regular expression.
	<b>show &lt;command&gt; include</b>	Filters <b>show</b> command output so that it displays only lines that contain a particular regular expression.

# show <command> append

To redirect and add the output of any **show** command to an existing file, use the **show command | append** command in privileged EXEC mode.

**show command | append URL**

Syntax Description	command	Any Cisco IOS <b>show</b> command.
	<b>  append URL</b>	<p>The addition of this syntax redirects the command output to the file location specified in the Universal Resource Locator (URL). The pipe ( ) is required.</p> <p>The Cisco IOS File System (IFS) uses URLs to specify the location of a file system, directory, and file. Typical URL elements include:</p> <p><i>prefix:[directory]/filename</i></p> <p>Prefixes can be local file locations, such as <b>flash:</b> or <b>disk0:</b>. Alternatively, you can specify network locations using the following syntax:</p> <p><b>ftp:</b><i>[[//[username[:password]@]location]/directory]/filename</i></p> <p><b>tftp:</b><i>[[//location]/directory]/filename</i></p> <p>The <b>rcp:</b> prefix is not supported.</p>

Command Modes	Privileged EXEC
---------------	-----------------

Command History	Release	Modification
	12.0(21)S	This command was introduced.
	12.2(13)T	This command was integrated into Cisco IOS Release 12.2 T.

Usage Guidelines	<p>To display all URL prefixes that are supported for this command, use the <b>show command   append ?</b> command.</p> <p>This command adds the <b>show</b> command output to the end of the specified file.</p>
------------------	---

Examples	<p>In the following example, output from the <b>show tech-support</b> command is redirected to an existing file on Disk 1 with the file-name of “showoutput.txt.” This output is added at the end of any existing data in the file.</p>
----------	---

```
Router# show tech-support | append disk1:showoutput.txt
```

Related Commands	Command	Description
	<a href="#">show &lt;command&gt; redirect</a>	Redirects the output of any <b>show</b> command to a specified file.
	<a href="#">show &lt;command&gt; tee</a>	Copies the <b>show</b> command output to a file while displaying it on the terminal.

■ show <command> begin

## show <command> begin

To begin the output of any **show** command from a specified string, use the **show command / begin** command in EXEC mode.

**show command / begin** *regular-expression*

Syntax Description		
	<i>command</i>	Any supported <b>show</b> command.
		A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
	<i>regular-expression</i>	Any regular expression found in <b>show</b> command output. The show output will begin from the first instance of this string (output prior to this string will not be printed to the screen). The string is case-sensitive. Use parenthesis to indicate a literal use of spaces.
	/	Specifies a search at a --More-- prompt that begins unfiltered output with the first line that contains the regular expression.
	-	Specifies a filter at a --More-- prompt that only displays output lines that do not contain the regular expression.
	+	Specifies a filter at a --More-- prompt that only displays output lines that contain the regular expression.

**Command Modes** EXEC

Command History	Release	Modification
	8.3	The <b>show</b> command was introduced.
	12.0(1)T	This extension of the <b>show</b> command was introduced.

**Usage Guidelines** The *regular-expression* argument is case sensitive and allows for complex matching requirements. Use parenthesis to indicate a literal use of spaces. For example, | **begin u** indicates that the show output should begin with any line that contains a u; | **begin ( u)** indicates that the show output should begin with any line that contains a space and a u together (line has a word that begins with a lowercase u).

To search the remaining output of the **show** command, use the following command at the --More-- prompt:

*/regular-expression*

You can specify a filtered search at any --More-- prompt. To filter the remaining output of the **show** command, use one of the following commands at the --More-- prompt:

*-regular-expression*

*+regular-expression*

When output volume is large, the search can produce long lists of output. To interrupt the output, press **Ctrl-^** (Ctrl-Shift-6) or **Ctrl-z**.

**Note**

Once you specify a filter for a **show** command, you cannot specify another filter at the next --More-- prompt. The first specified filter remains until the **more** command output finishes or until you interrupt the output. The use of the keyword **begin** does not constitute a filter.

Because prior output is not saved, you cannot search or filter backward through prior output.

**Note**

A few **show** commands that have long output requirements do not require user input at the --More-- prompt to jump to the next table of output; these types of output require you to enter the same number of Ctrl-^ or Ctrl-Z combinations as there are --More-- prompts to completely abort output.

**Examples**

The following is partial sample output of the **show interface | begin Ethernet** command that begins unfiltered output with the first line that contains the regular expression “Ethernet.” At the --More-- prompt, the user specifies a filter to show only the lines in the remaining output that contain the regular expression “Serial.”

```
router# show interface | begin Ethernet
Ethernet0 is up, line protocol is up
Hardware is Lance, address is 0060.837c.6399 (bia 0060.837c.6399)
  Description: ip address is 172.1.2.14 255.255.255.0
  Internet address is 172.1.2.14/24
.
.
.
      0 lost carrier, 0 no carrier
      0 output buffer failures, 0 output buffers swapped out
--More--
+Serial
filtering...
Serial1 is up, line protocol is up
Serial2 is up, line protocol is up
Serial3 is up, line protocol is down
Serial4 is down, line protocol is down
Serial5 is up, line protocol is up
Serial6 is up, line protocol is up
Serial7 is up, line protocol is up
```

**Related Commands**

Command	Description
<b>more &lt;url&gt; begin</b>	Begins unfiltered output of the <b>more</b> command with the first line that contains the regular expression you specify.
<b>more &lt;url&gt; exclude</b>	Filters <b>more</b> command output so that it excludes lines that contain a particular regular expression.
<b>more &lt;url&gt; include</b>	Filters <b>more</b> command output so that it displays only lines that contain a particular regular expression.
<b>show &lt;command&gt; exclude</b>	Filters <b>show</b> command output so that it excludes lines that contain a particular regular expression.
<b>show &lt;command&gt; include</b>	Filters <b>show</b> command output so that it displays only lines that contain a particular regular expression.

## show <command> exclude

To filter **show** command output so that it excludes lines that contain a particular regular expression, use the **show command | exclude** command in EXEC mode.

**show command | exclude regular-expression**

### Syntax Description

<i>command</i>	Any supported <b>show</b> command.
	A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
<i>regular-expression</i>	Any regular expression found in <b>show</b> command output.
/	Specifies a search at a --More-- prompt that begins unfiltered output with the first line that contains the regular expression.

### Command Modes

EXEC

### Command History

Release	Modification
8.3	The <b>show</b> command was introduced.
12.0(1)T	This extension of the <b>show</b> command was introduced.

### Usage Guidelines

The *regular-expression* argument is case sensitive and allows for complex matching requirements.

You can specify a new search at every --More-- prompt. To search the remaining output of the **show** command, use the following syntax at the --More-- prompt:

*/regular-expression*

When output volume is large, the search can produce long lists of output. To interrupt the output, press **Ctrl-^** (Ctrl-Shift-6) or **Ctrl-Z**.

Because prior output is not saved, you cannot search or filter backward through prior output.



#### Note

A few **show** commands that have long output requirements do not require user input at the --More-- prompt to jump to the next table of output; these types of output require you to enter the same number of Ctrl-^ or Ctrl-Z combinations as there are --More-- prompts to completely abort output.

### Examples

The following is partial sample output of the **show | exclude** command used with the **show buffers** command. It excludes lines that contain the regular expression “0 misses.” At the --More-- prompt, the user searches for the regular expression “Serial0,” which continues the filtered output with the first line that contains “Serial0.”

```
router# show buffers | exclude 0 misses

Buffer elements:
  398 in free list (500 max allowed)
Public buffer pools:
```

```

Small buffers, 104 bytes (total 50, permanent 50):
    50 in free list (20 min, 150 max allowed)
    551 hits, 3 misses, 0 trims, 0 created
Big buffers, 1524 bytes (total 50, permanent 50):
    49 in free list (5 min, 150 max allowed)
Very Big buffers, 4520 bytes (total 10, permanent 10):
.
.
.
Huge buffers, 18024 bytes (total 0 permanent 0):
    0 in free list (0 min, 4 max allowed)
--More--
/Serial0
filtering...
Serial0 buffers, 1543 bytes (total 64, permanent 64):
    16 in free list (0 min, 64 max allowed)
    48 hits, 0 fallbacks

```

---

**Related Commands**

Command	Description
<b>more begin</b>	Begins unfiltered output of the <b>more</b> command with the first line that contains the regular expression you specify.
<b>more exclude</b>	Filters <b>more</b> command output so that it excludes lines that contain a particular regular expression.
<b>more include</b>	Filters <b>more</b> command output so that it displays only lines that contain a particular regular expression.
<b>show &lt;command&gt; begin</b>	Searches the output of any <b>show</b> command and displays the output from the first instance of a specified string.
<b>show &lt;command&gt; include</b>	Filters <b>show</b> command output so that it displays only lines that contain a particular regular expression.

## show <command> include

To filter **show** command output so that it only displays lines that contain a particular regular expression, use the **show command | include** command in EXEC mode.

**show command | include** *regular-expression*

Syntax Description	<i>command</i>	Any supported <b>show</b> command.
		A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
	<i>regular-expression</i>	Any regular expression found in <b>show</b> command output. Use parenthesis to include spaces in the expression.
	/	Specifies a search at a --More-- prompt that begins unfiltered output with the first line that contains the regular expression.

Command Modes EXEC

Command History	Release	Modification
	8.3	The <b>show</b> command was introduced.
	12.0(1)T	This extension of the <b>show</b> command was introduced.

### Usage Guidelines

The *regular-expression* argument is case sensitive and allows for complex matching requirements.

You can specify a new search at every --More-- prompt. To search the remaining output of the **show** command, use the following syntax at the --More-- prompt:

*/regular-expression*

When output volume is large, the search can produce long lists of output. To interrupt the output, press **Ctrl-^** (Ctrl-Shift-6) or **Ctrl-Z**.

Because prior output is not saved, you cannot search or filter backward through prior output.



#### Note

A few **show** commands that have long output requirements do not require user input at the --More-- prompt to jump to the next table of output; these types of output require you to enter the same number of Ctrl-^ or Ctrl-Z combinations as there are --More-- prompts to completely abort output.

### Examples

The following is partial sample output of the **show interface | include ( is )** command. It displays only lines that contain the regular expression “( is ).” The parentheses force the inclusion of the spaces before and after “is.” Use of the parenthesis ensures that only lines containing “is” with a space both before and after it will be included in the output. Lines with words like “disconnect” will be excluded because there are not spaces around the instance of the string “is”.

```
router# show interface | include ( is )
```

```

ATM0 is administratively down, line protocol is down
  Hardware is ATMizer BX-50
Dialer1 is up (spoofing), line protocol is up (spoofing)
  Hardware is Unknown
  DTR is pulsed for 1 seconds on reset
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0060.837c.6399 (bia 0060.837c.6399)
  Internet address is 172.21.53.199/24
Ethernet1 is up, line protocol is up
  Hardware is Lance, address is 0060.837c.639c (bia 0060.837c.639c)
  Internet address is 5.5.5.99/24
Serial0:0 is down, line protocol is down
  Hardware is DSX1
.
.
.
--More--

```

At the --More-- prompt, the user searches for the regular expression “Serial0:13”, which continues filtered output with the first line that contains “Serial0:13.”

```

/Serial0:13
filtering...
Serial0:13 is down, line protocol is down
  Hardware is DSX1
  Internet address is 11.0.0.2/8
    0 output errors, 0 collisions, 2 interface resets
  Timeslot(s) Used:14, Transmitter delay is 0 flags

```

#### Related Commands

Command	Description
<a href="#">more &lt;url&gt; begin</a>	Begins unfiltered output of the <b>more</b> command with the first line that contains the regular expression you specify.
<a href="#">more &lt;url&gt; exclude</a>	Filters <b>more</b> command output so that it excludes lines that contain a particular regular expression.
<a href="#">more &lt;url&gt; include</a>	Filters <b>more</b> command output so that it displays only lines that contain a particular regular expression.
<a href="#">show &lt;command&gt; begin</a>	Searches the output of any <b>show</b> command and displays the output from the first instance of a specified string.
<a href="#">show &lt;command&gt; exclude</a>	Filters <b>show</b> command output so that it excludes lines that contain a particular regular expression.

## show <command> redirect

To redirect the output of any **show** command to a file, use the **show command | redirect** command in privileged EXEC mode.

**show command | redirect URL**

<b>Syntax Description</b>	<i>command</i>	Any Cisco IOS <b>show</b> command.
	<b>  redirect URL</b>	<p>The addition of this syntax redirects the command output to the file location specified in the Universal Resource Locator (URL). The pipe ( ) is required.</p> <p>The Cisco IOS File System (IFS) uses URLs to specify the location of a file system, directory, and file. Typical URL elements include:</p> <p><i>prefix:[directory]/filename</i></p> <p>Prefixes can be local file locations, such as <b>flash:</b> or <b>disk0:</b>. Alternatively, you can specify network locations using the following syntax:</p> <p><b>ftp:</b>[[//[username[:password]@]location]/directory]/filename</p> <p><b>tftp:</b>[[//location]/directory]/filename</p> <p>The <b>rcp:</b> prefix is not supported.</p>

<b>Command Modes</b>	Privileged EXEC
----------------------	-----------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.0(21)S	This command was introduced.
	12.2(13)T	This command was integrated into Cisco IOS Release 12.2 T.

<b>Usage Guidelines</b>	<p>To display all URL prefixes that are supported for this command, use the <b>show command   redirect ?</b> command.</p> <p>This command creates a new file at the specified location, or overwrites an existing file.</p>
-------------------------	---

<b>Examples</b>	<p>In the following example, output from the <b>show tech-support</b> command is written to the file “showtech.txt” on the host at 172.16.101.101 in the directory “//tftpboot/docs/” using FTP:</p> <pre>Rtr# show tech   redirect ftp://USER:MYPASSWORD@172.16.101.101//tftpboot/docs/showtech.txt</pre>
-----------------	--

---

Related Commands

Command	Description
<b>show &lt;command&gt; append</b>	Redirects and appends <b>show</b> command output to the end of an existing file.
<b>show &lt;command&gt; tee</b>	Copies the <b>show</b> command output to a file while displaying it on the terminal.

---

## show <command> tee

To copy the output of any **show** command to a file while displaying it on the terminal, use the **show command | tee** command in privileged EXEC mode.

```
show command | tee [/append] URL
```

Syntax Description	command	Any Cisco IOS <b>show</b> command.
	tee URL	The addition of this syntax copies the command output to the file location specified in the Universal Resource Locator (URL). The pipe ( ) is required.  The Cisco IOS File System (IFS) uses URLs to specify the location of a file system, directory, and file. Typical URL elements include:  <i>prefix:[directory]/filename</i>  Prefixes can be local file locations, such as <b>flash:</b> or <b>disk0:</b> . Alternatively, you can specify network locations using the following syntax:  <b>ftp:</b> [[//[username[:password]@]location]/directory]/filename <b>tftp:</b> [[//location]/directory]/filename  The <b>rcp:</b> prefix is not supported.
	/append	(Optional) Adds the <b>show</b> command output to the end of an existing file.

Command Modes	Privileged EXEC
---------------	-----------------

Command History	Release	Modification
	12.0(21)S	This command was introduced.
	12.2(13)T	This command was integrated into Cisco IOS Release 12.2 T.

Usage Guidelines	To display all URL prefixes that are supported for this command, use the <b>show command   tee ?</b> command.
------------------	---

The **tee** keyword was chosen to reflect that output is redirected to two locations; the terminal and a file (as a tee plumbing junction redirects water to two different pipes).

Examples	In the following example, output from the show tech-support command is displayed on-screen while it is written to the file “showoutput.txt” at the host 172.16.101.101 using TFTP:
----------	--

```
Router# show tech-support | tee tftp://172.16.101.101/docs/showoutput.txt
```

The following example performs the same function as above, but in this case the output is added at the end of any existing data in the file “showoutput.txt”:

```
Router# show tech-support | tee /append tftp://172.16.101.101/docs/showoutput.txt
```

---

**Related Commands**

Command	Description
<b>show &lt;command&gt; append</b>	Redirects the output of any <b>show</b> command and adds it to the end of existing file.
<b>show &lt;command&gt; redirect</b>	Redirects the output of any <b>show</b> command to a specified file.

# show history

To list the commands you have entered in the current EXEC session, use the **show history** command in EXEC mode.

**show history**

**Syntax Description** This command has no arguments or keywords.

**Command Modes** EXEC

Command History	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** The command history feature provides a record of EXEC commands you have entered. The number of commands that the history buffer will record is determined by the **history size** line configuration command or the **terminal history size** EXEC command.

[Table 3](#) lists the keys and functions you can use to recall commands from the command history buffer.

**Table 3 History Keys**

Key	Function
Ctrl-P or Up Arrow <sup>1</sup>	Recalls commands in the history buffer in a backward sequence, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or Down Arrow <sup>1</sup>	Returns to more recent commands in the history buffer after recalling commands with Ctrl-P or the Up Arrow. Repeat the key sequence to recall successively more recent commands.

1. The arrow keys function only with ANSI-compatible terminals.

**Examples** The following is sample output from the **show history** command, which lists the commands the user has entered in EXEC mode for this session:

```
Router# show history
  help
  where
  show hosts
  show history
Router#
```

Related Commands	Command	Description
	<b>history size</b>	Enables the command history function, or changes the command history buffer size for a particular line.
	<b>terminal history size</b>	Enables the command history feature for the current terminal session, or changes the size of the command history buffer for the current terminal session.

# terminal editing

To reenble the enhanced editing mode for only the current terminal session, use the **terminal editing** command in EXEC mode. To disable the enhanced editing mode on the current line, use the **no** form of this command.

**terminal editing**

**terminal no editing**

---

**Syntax Description** This command has no arguments or keywords.

---

**Defaults** Enabled

---

**Command Modes** EXEC

---

Command History	Release	Modification
	10.0	This command was introduced.

---



---

**Usage Guidelines** This command is identical to the **editing** EXEC mode command, except that it controls (enables or disables) enhanced editing for only the terminal session you are using. For a description of the available editing keys, see the description of the **editing** command in this chapter.

---

**Examples** In the following example, enhanced editing mode is reenbled for only the current terminal session:

```
Router> terminal editing
```

---

Related Commands	Command	Description
	<b>editing</b>	Controls CLI enhanced editing features for a particular line.

---

# terminal full-help

To get help for the full set of user-level commands, use the **terminal full-help** command in EXEC mode.

## terminal full-help

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** EXEC

Command History	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** The **terminal full-help** command enables a user to see all of the help messages available from the terminal. It is used with the **show ?** command.

**Examples** In the following example, the difference between the output of the **show ?** command before and after using the **terminal full-help** command is shown:

```
Router> show ?

bootflash  Boot Flash information
calendar   Display the hardware calendar
clock      Display the system clock
context    Show context information
dialer     Dialer parameters and statistics
history    Display the session command history
hosts      IP domain-name, lookup style, nameservers, and host table
isdn       ISDN information
kerberos   Show Kerberos Values
modemcap   Show Modem Capabilities database
ppp        PPP parameters and statistics
rmon       rmon statistics
sessions   Information about Telnet connections
snmp       snmp statistics
terminal   Display terminal configuration parameters
users      Display information about terminal lines
version    System hardware and software status

Router> terminal full-help
Router> show ?

access-expression  List access expression
access-lists       List access lists
aliases            Display alias commands
apollo            Apollo network information
```

```

appletalk      AppleTalk information
arp            ARP table
async         Information on terminal lines used as router interfaces
bootflash     Boot Flash information
bridge        Bridge Forwarding/Filtering Database [verbose]
bsc           BSC interface information
bstun         BSTUN interface information
buffers       Buffer pool statistics
calendar      Display the hardware calendar
cdp           CDP information
clns          CLNS network information
clock         Display the system clock
cls           DLC user information
cmns          Connection-Mode networking services (CMNS) information
compress      Show compression statistics.
.
.
.
x25           X.25 information
xns           XNS information
xremote       XRemote statistics

```

---

**Related Commands**

Command	Description
<a href="#">full-help</a>	Gets help for the full set of user-level commands.
<a href="#">help</a>	Displays a brief description of the help system.

---

# terminal history

To enable the command history feature for the current terminal session, use the **terminal history** command in user EXEC mode or privileged EXEC mode. To disable the command history feature, use the **no** form of this command.

**terminal history**

**terminal no history**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Enabled, history buffer of 10 lines

**Command Modes** EXEC

Command History	Release	Modification
	10.0	This command was introduced.

**Usage Guidelines** The history feature provides a record of commands you have entered. This feature is particularly useful for recalling long or complex commands or entries for the purposes of modifying them slightly and reexecuting them.

The **terminal history** command enables the command history feature with the default buffer size or the last buffer size specified using the **terminal history size** command.

[Table 4](#) lists the keys and functions you can use to recall commands from the history buffer.

**Table 4 History Keys**

Key(s)	Function
Ctrl-P or Up Arrow <sup>1</sup>	Recalls commands in the history buffer in a backward sequence, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or Down Arrow <sup>1</sup>	Returns to more recent commands in the history buffer after recalling commands with Ctrl-P or the Up Arrow. Repeat the key sequence to recall successively more recent commands.

1. The arrow keys function only with ANSI-compatible terminals.

**Examples** In the following example, the command history feature is disabled for the current terminal session:

```
Router> terminal no history
```

Related Commands	Command	Description
	<a href="#">history</a>	Enables the command history function, or changes the command history buffer size for a particular line.
	<a href="#">show history</a>	Lists the commands you have entered in the current EXEC session.
	<a href="#">terminal history size</a>	Sets the size of the history buffer for the command history feature for the current terminal session.

# terminal history size

To change the size of the command history buffer for the current terminal session, use the **terminal history size** command in EXEC mode. To reset the command history buffer to its default size of 10 lines, use the **no** form of this command.

**terminal history size** *number-of-lines*

**terminal no history size**

<b>Syntax Description</b>	<i>number-of-lines</i>	Number of command lines that the system will record in its history buffer. The range is from 0 to 256. The default is 10.
---------------------------	------------------------	---

<b>Defaults</b>	10 lines of command history
-----------------	-----------------------------

<b>Command Modes</b>	EXEC
----------------------	------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	10.0	This command was introduced.

**Usage Guidelines** The history feature provides a record of commands you have entered. This feature is particularly useful for recalling long or complex commands or entries for the purposes of modifying them slightly and reissuing them.

The **terminal history size** command enables the command history feature and sets the command history buffer size. The **terminal no history size** command resets the buffer size to the default of 10 command lines.

[Table 4](#) lists the keys and functions you can use to recall commands from the history buffer. When you use these keys, the commands recalled will be from EXEC mode if you are in EXEC mode, or from all configuration modes if you are in any configuration mode.

**Table 5 History Keys**

Key	Function
Ctrl-P or Up Arrow <sup>1</sup>	Recalls commands in the history buffer in a backward sequence, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or Down Arrow <sup>1</sup>	Returns to more recent commands in the history buffer after recalling commands with Ctrl-P or the Up Arrow. Repeat the key sequence to recall successively more recent commands.

1. The arrow keys function only with ANSI-compatible terminals.

In EXEC mode, you can also use the **show history** command to show the contents of the command history buffer.

To check the current settings for the command history feature on your line, use the **show line** command.

## Examples

In the following example, the number of command lines recorded is set to 15 for the current terminal session. The user then checks to see what line he/she is connected to using the **show users** command. The user uses this line information to issue the show line command. (In this example, the user uses the **show begin** option in the **show line** command to start the output at the “Editing is enabled/disabled” line.)

```
Router# terminal history size 15
Router# show users

      Line      User      Host(s)      Idle      Location
* 50 vty 0      admin      idle         00:00:00
! the * symbol indicates the active terminal session for the user (line 50)

Router# show line 50 | begin Editing

Editing is enabled.
! the following line shows the history settings for the line
History is enabled, history size is 15.
DNS resolution in show commands is enabled
Full user help is disabled
Allowed transports are telnet. Preferred is none.
No output characters are padded
No special data dispatching characters
```

## Related Commands

Command	Description
<a href="#">history</a>	Enables the command history function, or changes the command history buffer size for a particular line.
<a href="#">show &lt;command&gt; begin</a>	Searches the output of any show command and displays the output from the first instance of a specified string.
<a href="#">show history</a>	Lists the commands you have entered in the current EXEC session.
<a href="#">terminal history</a>	Enables the command history feature for the current terminal session.