



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 the **help** command and help features, lists the command editing keys and functions, and details the command history feature.

You can abbreviate the syntax of 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*.

disable

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

disable [*level*]

Syntax Description	<i>level</i> (Optional) Specifies the user-privilege level.
---------------------------	-------------------------------------------------------------

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

Command History	Release	Modification
	10.0	This command was introduced.

Usage Guidelines Up to 16 security levels can be configured using Cisco IOS. If such levels are configured on a system, using this command with the *level* option allows a user 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

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

Examples The following example shows use of the **disable** command to exit from privileged EXEC mode (indicated by a # after the router prompt) to user EXEC mode (indicated by a > router prompt):

```
Router# disable
Router>
```

Related Commands	Command	Description
	enable	Enables higher privilege level access, such as privileged EXEC mode.

editing

To disable Cisco IOS enhanced editing features, use the **no** form of this line configuration command. To reenable these features for a particular line, use the **editing** form of this command.

editing

no editing

Syntax Description This command has no arguments or keywords.

Defaults Enabled

Command Modes Line configuration

Command History	Release	Modification
	10.0	This command was introduced.

Usage Guidelines Enhanced editing features are, by default, enabled on the Cisco IOS. However, there may be situations in which a user may want 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 3 provides a description of the keys used to enter and edit commands when the editing features are enabled. Ctrl indicates the Control key. It must be pressed simultaneously with its associated letter key. Esc indicates the Escape key. It must be pressed first, followed by its associated letter key. Keys are case sensitive.

Table 3 *Command Editing Keys and Functions*

Keys	Function
Tab	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. Enter a question mark (?) immediately following the partial command (no space). The system provides a list of commands that begin with that string.
Delete or Backspace	Erases the character to the left of the cursor.
Return	At the command line, pressing the Return key performs the function of processing a command. At the More prompt on a terminal screen, pressing the Return key scrolls down a line.
Space Bar	Allows you to see more output on the terminal screen. Press the space bar when you see the More prompt on the screen to display the next screen.

Table 3 Command Editing Keys and Functions (continued)

Keys	Function
Left Arrow ¹	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 key repeatedly to scroll back toward the system prompt and verify the beginning of the command entry.
Right Arrow ¹	Moves the cursor one character to the right.
Up Arrow ¹ or Ctrl-P	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down Arrow ¹ or Ctrl-N	Return 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-A	Moves the cursor to the beginning of the line.
Ctrl-B	Moves the cursor back one character.
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Moves the cursor to the end of the command line.
Ctrl-F	Moves the cursor forward one character.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-L and Ctrl-R	Redisplays the system prompt and command line.
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.
Ctrl-U and Ctrl-X	Deletes all characters from the cursor back to the beginning of the command line.
Ctrl-V and Esc Q	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.
Ctrl-W	Deletes the word to the left of the cursor.
Ctrl-Y	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.
Ctrl-Z	Ends configuration mode and returns you to the EXEC prompt.
Esc B	Moves the cursor back one word.
Esc C	Capitalizes the word from the cursor to the end of the word.
Esc D	Deletes from the cursor to the end of the word.
Esc F	Moves the cursor forward one word.
Esc L	Changes the word to lowercase at the cursor to the end of the word.

Table 3 *Command Editing Keys and Functions (continued)*

Keys	Function
Esc U	Capitalizes from the cursor to the end of the word.
Esc Y	Recalls the next buffer entry. The 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.

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

Examples

The following example displays an enhanced editing mode disabled on line 3:

```
line 3
no editing
```

Related Commands

Command	Description
terminal editing	Enables the enhanced editing mode on the local line.

enable

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

```
enable [level]
```

Syntax Description

level (Optional) Privileged level on which to log in.

Command Modes

EXEC

Command History

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 from the router console.

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 traditionally associated with normal EXEC mode, and privilege level 15 is traditionally 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).



Note

The **enable** command is associated with privilege level 0. If you configure AAA authorization for a privilege level greater than 0, this command will not be included in the command set for that privilege level.

Examples

The following example displays the **enable** command being entered and a prompt to enter a password. The password is not displayed on the screen. After the user enters the correct password, the system enters privileged command mode, as indicated by the pound sign (#).

```
Router> enable
Password:
Router#
```

Related Commands	Command	Description
	disable	Exits the user to lower privilege levels, such as returning to user EXEC mode from privileged EXEC mode.
	enable password	Sets a local password to control access to various privilege levels.

end

To exit configuration mode, or any of the configuration submodes, use the **end** global configuration command.

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 You can also press **Ctrl-Z** to exit configuration mode.

Examples The following example changes the name to george using the **hostname** global configuration command. Entering the **end** command causes the system to exit configuration mode and return to privileged EXEC mode.

```
Router(config)# hostname george
george(config)# end
george#
```

Related Commands	Command	Description
	hostname	Specifies or modify the host name for the network server.

exit

To exit any configuration mode or close an active terminal session and terminate the EXEC, use the **exit** command at the system prompt.

exit

Syntax Description This command has no arguments or keywords.

Defaults No default behavior or values

Command Modes Available in all command modes.

Command History	Release	Modification
	10.0	This command was introduced.

Usage Guidelines Use the **exit** command at the EXEC levels to exit the EXEC mode. Use the **exit** command at the configuration level to return to privileged EXEC mode. Use the **exit** command in interface, line, router, IPX-router, and route-map command modes to return to global configuration mode. Use the **exit** command in subinterface configuration mode to return to interface configuration mode. You also can press **Ctrl-Z**, or use the **end** command, from any configuration mode to return to privileged EXEC mode.



Note

The **exit** command is associated with privilege level 0. If you configure AAA authorization for a privilege level greater than 0, this command will not be included in the command set for that privilege level.

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)#
```

The following example shows how to exit an active session (log-off):

```
Router> exit
```

Related Commands

Command	Description
disconnect	Disconnects a line.
end	Exits configuration mode, or any of the configuration submodes.

full-help

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

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 The following example is output for the **show ?** command with **full-help** disabled and then enabled:

```
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:
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)# end
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
help	Displays a brief description of the help system.

help

To display a brief description of the help system, enter the **help** command.

help

Syntax Description This command has no arguments or keywords.

Defaults No default behavior or values

Command Modes Available in 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.

- 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 a command's associated keywords or arguments, 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.



Note

The **help** command is associated with privilege level 0. If you configure AAA authorization for a privilege level greater than 0, this command will not be included in the command set for that privilege level.

Examples

The following example displays the **help** command giving 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”:

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

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.

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

Related Commands

Command	Description
full-help	Gets help for the full set of user-level commands.

history

To enable the command history function, use the **history** line configuration command. 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, 10 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** 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 10 lines. The **no history** command disables the history feature.

The **show history** command will list the commands you have entered, but you can also use your keyboard to display individual commands. Table 4 lists the keys you can use to recall commands from the command history buffer.

Table 4 History Keys

Key	Functions
Ctrl-P or Up Arrow ¹	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 ¹	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 such as VT100s.

Examples

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

```
line 4
  history size 35
```

Related Commands

Command	Description
history size	Sets the command history buffer size for a particular line.
show history	Lists the commands you have entered in the current EXEC session.
terminal history	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** line configuration command. 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

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

Defaults	Ten command lines
-----------------	-------------------

Command Modes	Line configuration
----------------------	--------------------

Command History	Release	Modification
	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:

```
line 4
 history size 35
```

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

logout

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

logout

Syntax Description This command has no arguments or keywords.

Defaults No default behavior or values.

Command Modes User EXEC

Command History	Release	Modification
	10.0	This command was introduced.

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 **logout** command is used to log off (exit from the active session):

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

Related Commands	Command	Description
	exit (global)	Exits any configuration mode to the next highest mode in the CLI mode hierarchy.

menu (EXEC)

To display a preconfigured user menu, use the **menu** EXEC command.

menu *name*

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

Command Modes	User EXEC or privileged EXEC
----------------------	------------------------------

Command History	Release	Modification
	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, which the user can use 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 table below. 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
menu command	Specifies underlying commands for user interface menus.
menu clear-screen	Clears the terminal screen before displaying a menu.
menu default	Specifies the menu item to use as the default.
menu-exit	Specifies an exit option in a user menu. See the menu command command for details.
menu line-mode	Requires the user to press Enter after specifying an option number.
menu options	Sets options for items in user interface menus.
menu prompt	Specifies the prompt for a user interface menu.
menu single-space	Displays menu items single-spaced rather than double-spaced.
menu status-line	Displays a line of status information about the current user at the top of a menu
menu text	Specifies the text of a menu item in a user interface menu.
menu title	Creates a title, or banner, for a user menu.
no menu	Deletes a specified menu from a menu configuration.

more begin

To search the output of any **more** command, use the **more 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 *any-command* | **begin** *regular-expression*

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

Syntax Description

<i>any-command</i>	Any supported more command.
	A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
<i>regular-expression</i>	Any regular expression found in more 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
12.0(1)T	This modification of the more 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.



Note

You can only specify one filter for each **more** command. The filter remains until the **more** command output finishes or until you interrupt the output (using **Ctrl-^**).

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

When output volume is large, the search can produce long lists of output. To interrupt this output, press **Ctrl-^**.

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
more exclude	Filters more command output so that it excludes lines that contain a particular regular expression.
more include	Filters more command output so that it only displays lines that contain a particular regular expression.
show begin	Searches the output of any show command.
show exclude	Filters show command output so that it excludes lines that contain a particular regular expression.
show include	Filters show command output so that it only displays lines that contain a particular regular expression.

more 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 any-command | exclude regular-expression
```

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

```
/regular-expression
```

Syntax Description

<i>any-command</i>	Any supported more command.
	A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
<i>regular-expression</i>	Any regular expression found in more 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
12.0(1)T	This modification of the more 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.



Note

Once you specify a filter for a **more** command, you cannot specify another filter at a --More-- prompt. The filter remains until the **more** command output finishes or until you interrupt the output (using **Ctrl-^**).

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

When output volume is large, the search can produce long lists of output. To interrupt this output, press **Ctrl-^**.

Examples

The following is partial sample output of the **more nvram:startup-config | exclude service** command. It excludes lines that contain the regular expression “service.” At the --More-- prompt, the user searches for the regular expression “Dialer1.” This 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
more begin	Begins unfiltered output of the more command with the first line that contains the regular expression you specify.
more include	Filters more command output so that it only displays lines that contain a particular regular expression.
show begin	Searches the output of any show command.
show exclude	Filters show command output so that it excludes lines that contain a particular regular expression.
show include	Filters show command output so that it only displays lines that contain a particular regular expression.

more include

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

more *any-command* | **include** *regular-expression*

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

/regular-expression

Syntax Description

<i>any-command</i>	Any supported more command.
	A vertical bar (the “ ” symbol) indicates that an output processing specification follows.
<i>regular-expression</i>	Any regular expression found in more 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
12.0(1)T	This modification of the more 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.



Note

Once you specify a filter for a **more** command, you cannot specify another filter at a --More-- prompt. The filter remains until the **more** command output finishes or until you interrupt the output (using **Ctrl-^**).

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

When output volume is large, the search can produce long lists of output. To interrupt this output, press **Ctrl-^**.

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
more begin	Begins unfiltered output of the more command with the first line that contains the regular expression you specify.
more exclude	Filters more command output so that it excludes lines that contain a particular regular expression.
show begin	Searches the output of any show command.
show exclude	Filters show command output so that it excludes lines that contain a particular regular expression.
show include	Filters show command output so that it only displays lines that contain a particular regular expression.

show begin

To search the output of any **show** command, use the **show begin** command in EXEC mode.

```
show any-command | begin regular-expression
```

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

```
/regular-expression
```

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

```
-regular-expression
```

```
+regular-expression
```

Syntax Description		
	<i>any-command</i>	Any supported show command.
		A vertical bar (the “pipe” symbol) indicates that an output processing specification follows.
	<i>regular-expression</i>	Any regular expression found in show 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
	12.0(1)T	This modification of the show 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.



Note

You can only specify one filter for each **show** command. The filter remains until the **show** command output finishes or until you interrupt the output (using **Ctrl-^**).

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

When output volume is large, the search can produce long lists of output. To interrupt this output, press **Ctrl-^**.

**Note**

A few **show** commands that have long output requirements use no responses at the --More-- prompt to jump to the next table of output; these outputs require you to enter the same number of Ctrl-^ combinations as you would no responses 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 include only the lines 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
more begin	Begins unfiltered output of the more command with the first line that contains the regular expression you specify.
more exclude	Filters more command output so that it excludes lines that contain a particular regular expression.
more include	Filters more command output so that it only displays lines that contain a particular regular expression.
show exclude	Filters show command output so that it excludes lines that contain a particular regular expression.
show include	Filters show command output so that it only displays lines that contain a particular regular expression.

show exclude

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

```
show any-command | exclude regular-expression
```

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

```
/regular-expression
```

Syntax Description

<i>any-command</i>	Any supported show command.
	A vertical bar (the “ ” symbol) indicates that an output processing specification follows.
<i>regular-expression</i>	Any regular expression found in show 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
12.0(1)T	This modification of the show 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.



Note

Once you specify a filter for a **show** command, you cannot specify another filter at a --More-- prompt. The filter remains until the **show** command output finishes or until you interrupt the output (using **Ctrl-^**).

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

When output volume is large, the search can produce long lists of output. To interrupt this output, press **Ctrl-^**.



Note

A few **show** commands that have long output requirements use no responses at the --More-- prompt to jump to the next table of output; these outputs require you to enter the same number of **Ctrl-^** combinations as you would no responses 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.” This 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
more begin	Begins unfiltered output of the more command with the first line that contains the regular expression you specify.
more exclude	Filters more command output so that it excludes lines that contain a particular regular expression.
more include	Filters more command output so that it only displays lines that contain a particular regular expression.
show begin	Searches the output of any show command.
show include	Filters show command output so that it only displays lines that contain a particular regular expression.

show history

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

show history

Syntax Description This command has no arguments or keywords.

Command Modes EXEC

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 5 lists the keys and functions you can use to recall commands from the command history buffer.

Table 5 History Keys

Key	Function
Ctrl-P or Up Arrow	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	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.

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#
```

■ show history

Related Commands

Command	Description
history size	Enables the command history function, or changes the command history buffer size for a particular line.
terminal history size	Enables the command history feature for the current terminal session or change the size of the command history buffer for the current terminal session.

show include

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

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

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

/regular-expression

Syntax Description

<i>any-command</i>	Any supported show command.
	A vertical bar (the “ ” symbol) indicates that an output processing specification follows.
<i>regular-expression</i>	Any regular expression found in show 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
12.0(1)T	This modification of the show 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.



Note

Once you specify a filter for a **show** command, you cannot specify another filter at a --More-- prompt. The filter remains until the **show** command output finishes or until you interrupt the output (using **Ctrl-^**).

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

When output volume is large, the search can produce long lists of output. To interrupt this output, press **Ctrl-^**.



Note

A few **show** commands that have long output requirements use no responses at the --More-- prompt to jump to the next table of output; these outputs require you to enter the same number of **Ctrl-^** combinations as you would no responses to completely abort output.

Examples

The following is partial sample output of the **show interface | include (is)** command. It only displays lines that contain the regular expression “(is).” The parentheses force the inclusion of the spaces before and after “is.” This ensures that only lines containing “is” with a space both before and after it will be included in the output. This excludes lines with words like “disconnect.”

```
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.” This 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
more begin	Begins unfiltered output of the more command with the first line that contains the regular expression you specify.
more exclude	Filters more command output so that it excludes lines that contain a particular regular expression.
more include	Filters more command output so that it only displays lines that contain a particular regular expression.
show begin	Searches the output of any show command.
show exclude	Filters show command output so that it excludes lines that contain a particular regular expression.

terminal editing

To enable the enhanced editing mode on the local line, use the **terminal editing** EXEC command. 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 Table 6 provides a description of the keys used to enter and edit commands. Ctrl indicates the Control key. It must be pressed simultaneously with its associated letter key. Esc indicates the Escape key. It must be pressed first, followed by its associated letter key. Keys are *not* case sensitive.

Table 6 Command Editing Keys and Functions

Keys	Function
Tab	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. Enter a question mark (?) immediately following the partial command (no space). The system provides a list of commands that begin with that string.
Delete or Backspace	Erases the character to the left of the cursor.
Return	At the command line, pressing the Return key performs the function of processing, or carrying out, a command. At the More prompt on a terminal screen, pressing the Return key scrolls down a line.
Space Bar	Scrolls down a page on the terminal screen. Press the space bar when you see the More prompt on the screen to display the next screen.
Left arrow ¹	Moves the cursor one character to the left. When you enter a command that extends beyond a single line, you can continue to press the left arrow key at any time to scroll back toward the system prompt and verify the beginning of the command entry.

Table 6 Command Editing Keys and Functions (continued)

Keys	Function
Right arrow ¹	Moves the cursor one character to the right.
Up arrow ¹ or Ctrl-P	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down arrow ¹ or Ctrl-N	Return 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-A	Moves the cursor to the beginning of the line.
Ctrl-B	Moves the cursor back one character.
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Moves the cursor to the end of the command line.
Ctrl-F	Moves the cursor forward one character.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-L and Ctrl-R	Redisplays the system prompt and command line.
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.
Ctrl-U and Ctrl-X	Deletes all characters from the cursor back to the beginning of the command line.
Ctrl-V and Esc Q	Inserts a code to indicate to the system that the key stroke immediately following should be treated as a command entry, <i>not</i> as an editing key.
Ctrl-W	Deletes the word to the left of the cursor.
Ctrl-Y	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.
Ctrl-Z	Ends configuration mode and returns you to the EXEC prompt.
Esc B	Moves the cursor back one word.
Esc C	Capitalizes the word at the cursor.
Esc D	Deletes from the cursor to the end of the word.
Esc F	Moves the cursor forward one word.
Esc L	Changes the word at the cursor to lowercase.
Esc U	Capitalizes from the cursor to the end of the word.
Esc Y	Recalls the next buffer entry. The 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.

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

Examples

The following example displays enhanced mode editing reenabled for the current terminal session:

```
terminal editing
```

Related Commands

Command	Description
editing	Reenables enhanced editing mode for a particular line.

terminal full-help

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

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 (or disables) a user to see all of the help messages available from the terminal. It is used with the **show ?** command.

Examples The following example displays output for the **show ?** command with **terminal full-help** disabled and then enabled:

```
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
full-help	Gets help for the full set of user-level commands.
help	Displays a brief description of the help system.

terminal history

To enable the command history feature for the current terminal session or change the size of the command history buffer for the current terminal session, use the **terminal history** EXEC command. To disable the command history feature or reset the command history buffer to its default size, use the **no** form of this command.

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

terminal no history [*size*]

Syntax Description	size	(Optional) Sets command history buffer size.
	<i>number-of-lines</i>	(Optional) Specifies the number of command lines that the system will record in its history buffer. The range is 0 to 256. The default is 10.

Defaults	10 lines
----------	----------

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

Command History	Release	Modification
	10.0	This command was introduced.

Usage Guidelines

The **terminal history** command without the **size** keyword and argument enables the command history feature with the last buffer size specified or the default size. The **terminal no history** command without the **size** keyword disables the command history feature. The **terminal no history size** command resets the buffer size to the default of 10 command lines.

The **terminal history** command provides a record of EXEC commands you have entered. This feature is particularly useful to recall long or complex commands or entries, including access lists.

Table 7 lists the keys and functions you can use to recall commands from the history buffer.

Table 7 History Keys

Key	Function
Ctrl-P or Up Arrow ¹	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 ¹	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 such as VT100s.

Examples

The following example displays the number of command lines recorded is set to 15 for the local line:

```
terminal history size 15
```

Related Commands

Command	Description
history	Enables the command history function, or changes the command history buffer size for a particular line.
show history	Lists the commands you have entered in the current EXEC session.

