

Use Cisco IOS XE software

This chapter provides information to prepare you to configure the Cisco C8400 Series Secure Router :

- Limitations for Cisco IOS XE software, on page 1
- Access the CLI using a router console, on page 1
- Understand command mode, on page 2
- Get help, on page 3
- Use the no and default forms of commands, on page 7
- Save configuration changes, on page 7
- Managing Configuration Files, on page 7
- Dynamic allocation of cores, on page 9
- Filter the output of the show and more commands, on page 9
- Disable front-panel USB ports, on page 10
- Power off a router, on page 11

Limitations for Cisco IOS XE software

The maximum supported size of a Cisco IOS XE software image file is 4 GB.

Access the CLI using a router console

The following sections describe how to access the command-line interface (CLI) using a directly-connected console or by using Telnet or a modem to obtain a remote console:

Access the CLI using a directly-connected console

This section describes how to connect to the console port on the router and use the console interface to access the CLI.

The console port on a Cisco C8400 Series Secure Router is an EIA/TIA-232 asynchronous, serial connection with no flow control and an RJ-45 connector. The console port is located on the front panel of each router.

Connect to the router console using telnet

Before you can access the router remotely using Telnet from a TCP/IP network, you need to configure the router to support virtual terminal lines (vtys) using the **line vty** global configuration command. You also should configure the vtys to require login and specify a password.

Note To prevent disabling login on the line, be careful that you specify a password with the **password** command when you configure the **login** line configuration command. If you are using authentication, authorization, and accounting (AAA), you should configure the **login authentication** line configuration command. To prevent disabling login on the line for AAA authentication when you configure a list with the **login authentication** command, you must also configure that list using the **aaa authentication login** global configuration command. For more information about AAA services, see the *Cisco IOS XE Security Configuration Guide,* and the *Cisco IOS Security Command Reference Guide* .

In addition, before you can make a Telnet connection to the router, you must have a valid host name for the router or have an IP address configured on the router. For more information about requirements for connecting to the router using Telnet, information about customizing your Telnet services, and using Telnet key sequences, see the *Cisco IOS Configuration Fundamentals Configuration Guide.*

Understand command mode

The command modes available in the traditional Cisco IOS CLI are exactly the same as the command modes available in Cisco IOS XE.

You use the CLI to access Cisco IOS XE software. Because the CLI is divided into many different modes, the commands available to you at any given time depend on the mode that you are currently in. Entering a question mark (**?**) at the CLI prompt allows you to obtain a list of commands available for each command mode.

When you log in to the CLI, you are in user EXEC mode. User EXEC mode contains only a limited subset of commands. To have access to all commands, you must enter privileged EXEC mode, normally by using a password. From privileged EXEC mode, you can issue any EXEC command—user or privileged mode—or you can enter global configuration mode. Most EXEC commands are one-time commands. For example, **show** commands show important status information, and **clear** commands clear counters or interfaces. The EXEC commands are not saved when the software reboots.

Configuration modes allow you to make changes to the running configuration. If you later save the running configuration to the startup configuration, these changed commands are stored when the software is rebooted. To enter specific configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and a variety of other modes, such as protocol-specific modes.

ROM monitor mode is a separate mode used when the Cisco IOS XE software cannot load properly. If a valid software image is not found when the software boots or if the configuration file is corrupted at startup, the software might enter ROM monitor mode.

The following table describes how to access and exit various common command modes of the Cisco IOS XE software. It also shows examples of the prompts displayed for each mode.

Table 1: Accessing and Exiting Command Modes

Command Mode	Access Method	Prompt	Exit Method
User EXEC	Log in.	`Router>`	Use the **logout** command.
Privileged EXEC	From user EXEC mode, use the **enable** EXEC command.	`Router#`	To return to user EXEC mode, use the **disable** command.
Global configuration	From privileged EXEC mode, use the **configure terminal** privileged EXEC command.	`Router(config)#`	To return to privileged EXEC mode from global configuration mode, use the **exit** or **end** command.
Interface configuration	From global configuration mode, specify an interface using an **interface** command.	`Router(config-if)#`	To return to global configuration mode, use the **exit** command. To return to privileged EXEC mode, use the **end** command.
Diagnostic	The router boots up or accesses diagnostic mode in the following scenarios: In some cases, diagnostic mode will be reached when the IOS process or processes fail. In most scenarios, however, the router will. A user-configured access policy was configured using the **transport-map** command that directed the user into diagnostic mode. See the Chapter 4, "Console Port, Telnet, and SSH Handling" of this book for information on configuring access policies. The router was accessed using a Route Processor auxiliary port. A break signal (**Ctrl-C**, **Ctrl-Shift-6**, or the **send break** command) was entered and the router was configured to go into diagnostic mode when the break signal was received.	`Router(diag)#`	If the IOS process failing is the reason for entering diagnostic mode, the IOS problem must be resolved and the router rebooted to get out of diagnostic mode. If the router is in diagnostic mode because of a transport-map configuration, access the router through another port or using a method that is configured to connect to the Cisco IOS CLI. If the router is accessed through the Route Processor auxiliary port, access the router through another port. Accessing the router through the auxiliary port is not useful for customer purposes anyway.
ROM monitor	From privileged EXEC mode, use the **reload** EXEC command. Press the **Break** key during the first 60 seconds while the system is booting.	`>`	To exit ROM monitor mode, use the **continue** command.

Get help

Entering a question mark (**?**) at the CLI prompt displays a list of commands available for each command mode. You can also get a list of keywords and arguments associated with any command by using the context-sensitive help feature.

To get help specific to a command mode, a command, a keyword, or an argument, use one of the commands listed in the following table:

Table 2: Help Commands and Purpose

Command	Purpose
help	Provides a brief description of the help system in any command mode.
abbreviated-command-entry?	Provides a list of commands that begin with a particular character string. (No space between command and question mark.)
abbreviated-command-entry<Tab>	Completes a partial command name.
?	Lists all commands available for a particular command mode.
command ?	Lists the keywords or arguments that you must enter next on the command line. (Space between command and question mark.)

Finding command options

This section provides an example of how to display syntax for a command. The syntax can consist of optional or required keywords and arguments. To display keywords and arguments for a command, enter a question mark (**?**) at the configuration prompt or after entering part of a command followed by a space. The Cisco IOS XE software displays a list and brief description of available keywords and arguments. For example, if you were in global configuration mode and wanted to see all the keywords or arguments for the **arap** command, you would type **arap ?**.

The <cr> symbol in command help output stands for "carriage return." On older keyboards, the carriage return key is the Return key. On most modern keyboards, the carriage return key is the Enter key. The <cr> symbol at the end of command help output indicates that you have the option to press **Enter** to complete the command and that the arguments and keywords in the list preceding the <cr> symbol are optional. The <cr> symbol by itself indicates that no more arguments or keywords are available and that you must press **Enter** to complete the command.

The following table shows examples of how you can use the question mark (**?**) to assist you in entering commands.

Table 3: Finding Command Options

Command	Comment

`Router> enable`
`Password: <password>`
`Router#`
	Enter the **enable** command and password to access privileged EXEC commands. You are in privileged EXEC mode when the prompt changes to a "# " from the "> "; for example, Router> to Router# .

`Router#`
`configure terminal`
`Enter configuration commands, one per line. End with CNTL/Z.`
`Router(config)#`
	Enter the **configure terminal** privileged EXEC command to enter global configuration mode. You are in global configuration mode when the prompt changes to Router(config)# .

Command	Comment
`Router(config)# interface GigabitEthernet 0/0/0 ?`	Enter interface configuration mode by specifying the serial interface that you want to configure using the **interface GigabitEthernet 0/0/0** global configuration command. Enter **?** to display what you must enter next on the command line. In this example, you must enter the serial interface slot number and port number, separated by a forward slash. When the <cr> symbol is displayed, you can press Enter to complete the command. You are in interface configuration mode when the prompt changes to Router(config-if)# .

`Router(config-if)# ?`
`Interface configuration commands:`
`  .`
`  .`
`  .`
`  ip  Interface Internet Protocol config commands`
`  keepalive  Enable keepalive`
`  lan-name  LAN Name command`
`  llc2  LLC2 Interface Subcommands`
`  load-interval  Specify interval for load calculation for an interface`
`  locaddr-priority  Assign a priority group`
`  logging  Configure logging for interface`
`  loopback  Configure internal loopback on an interface`
`  mac-address  Manually set interface MAC address`
`  mls  mls router sub/interface commands`
`  mpoa  MPOA interface configuration commands`
`  mtu  Set the interface Maximum Transmission Unit (MTU)`
`  netbios  Use a defined NETBIOS access list or enable name-caching`
`  no  Negate a command or set its defaults`
`  nrzi-encoding  Enable use of NRZI encoding`
`  ntp  Configure NTP`
`  .`
`  .`
`  .`
`Router(config-if)#`
	Enter **?** to display a list of all the interface configuration commands available for the serial interface. This example shows only some of the available interface configuration commands.

Command	Comment

`Router(config-if)# ip ?`
`Interface IP configuration subcommands:`
`  access-group  Specify access control for packets`
`  accounting  Enable IP accounting on this interface`
`  address  Set the IP address of an interface`
`  authentication  authentication subcommands`
`  bandwidth-percent  Set EIGRP bandwidth limit`
`  broadcast-address  Set the broadcast address of an interface`
`  cgmp  Enable/disable CGMP`
`  directed-broadcast  Enable forwarding of directed broadcasts`
`  dvmrp  DVMRP interface commands`
`  hello-interval  Configures IP-EIGRP hello interval`
`  helper-address  Specify a destination address for UDP broadcasts`
`  hold-time  Configures IP-EIGRP hold time`
`  .`
`  .`
`  .`
`Router(config-if)# ip`
	Enter the command that you want to configure for the interface. This example uses the **ip** command.
	Enter **?** to display what you must enter next on the command line. This example shows only some of the available interface IP configuration commands.

`Router(config-if)# ip address ?`
`  A.B.C.D  IP address`
`  negotiated  IP Address negotiated over PPP`
`Router(config-if)# ip address`
	Enter the command that you want to configure for the interface. This example uses the **ip address** command.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP address or the **negotiated** keyword.
	A carriage return (<cr>) is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

`Router(config-if)# ip address 172.16.0.1 ?`
`  A.B.C.D  IP subnet mask`
`Router(config-if)# ip address 172.16.0.1`
	Enter the keyword or argument that you want to use. This example uses the 172.16.0.1 IP address.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP subnet mask.
	A <cr> is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

`Router(config-if)# ip address 172.16.0.1 255.255.255.0 ?`
`  secondary  Make this IP address a secondary address`
`  <cr>`
`Router(config-if)# ip address 172.16.0.1 255.255.255.0`
	Enter the IP subnet mask. This example uses the 255.255.255.0 IP subnet mask.
	Enter **?** to display what you must enter next on the command line. In this example, you can enter the **secondary** keyword, or you can press **Enter**.
	A <cr> is displayed; you can press **Enter** to complete the command, or you can enter another keyword.

`Router(config-if)# ip address 172.16.0.1 255.255.255.0` `Router(config-if)#`	In this example, **Enter** is pressed to complete the command.

Use the no and default forms of commands

Almost every configuration command has a **no** form. In general, use the **no** form to disable a function. Use the command without the **no** keyword to re-enable a disabled function or to enable a function that is disabled by default. For example, IP routing is enabled by default. To disable IP routing, use the **no ip routing** command; to re-enable IP routing, use the **ip routing** command. The Cisco IOS software command reference publications provide the complete syntax for the configuration commands and describe what the **no** form of a command does.

Many CLI commands also have a **default** form. By issuing the command **default** *command-name* , you can configure the command to its default setting. The Cisco IOS software command reference publications describe the function of the **default** form of the command when the **default** form performs a different function than the plain and **no** forms of the command. To see what default commands are available on your system, enter **default ?** in the appropriate command mode.

Save configuration changes

Use the **copy running-config startup-config** command to save your configuration changes to the startup configuration so that the changes will not be lost if the software reloads or a power outage occurs. For example:

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

It might take a minute or two to save the configuration. After the configuration has been saved, the following output appears:

```
[OK]
Router#
```

This task saves the configuration to NVRAM.

Managing Configuration Files

On the Cisco C84XX Series Platforms , the startup configuration file is stored in the nvram: file system and the running-configuration files are stored in the system: file system. This configuration file storage setup is not unique to the Cisco C84XX Series Platforms and is used on several Cisco router platforms.

As a matter of routine maintenance on any Cisco router, users should backup the startup configuration file by copying the startup configuration file from NVRAM onto one of the router's other file systems and, additionally, onto a network server. Backing up the startup configuration file provides an easy method of recovering the startup configuration file in the event the startup configuration file in NVRAM becomes unusable for any reason.

The **copy** command can be used to backup startup configuration files. The following examples show the startup configuration file in NVRAM being backed up:

Example 1: Copying a Startup Configuration File to Bootflash

```
Router# dir bootflash:
```

```
Directory of bootflash:/
  11 drwx 16384 Sep 18 2020 15:16:35 +00:00 lost+found
1648321 drwx 4096 Oct 22 2020 12:08:47 +00:00 .installer
97921 drwx 4096 Sep 18 2020 15:18:00 +00:00 .rollback_timer
12 -rw- 1910 Oct 22 2020 12:09:09 +00:00 mode_event_log
1566721 drwx 4096 Sep 18 2020 15:33:23 +00:00 core
1215841 drwx 4096 Oct 22 2020 12:09:48 +00:00 .prst_sync
1289281 drwx 4096 Sep 18 2020 15:18:18 +00:00 bootlog_history
13 -rw- 133219 Oct 22 2020 12:09:34 +00:00 memleak.tcl
14 -rw- 20109 Sep 18 2020 15:18:39 +00:00 ios_core.p7b
15 -rwx 1314 Sep 18 2020 15:18:39 +00:00 trustidrootx3_ca.ca
391681 drwx 4096 Oct 6 2020 15:08:54 +00:00 .dbpersist
522241 drwx 4096 Sep 18 2020 15:32:59 +00:00 .inv
783361 drwx 49152 Oct 27 2020 08:36:44 +00:00 tracelogs
832321 drwx 4096 Sep 18 2020 15:19:17 +00:00 pnp-info
1207681 drwx 4096 Sep 18 2020 15:19:20 +00:00 onep
750721 drwx 4096 Oct 22 2020 12:09:57 +00:00 license_evlog
946561 drwx 4096 Sep 18 2020 15:19:24 +00:00 guest-share
383521 drwx 4096 Sep 18 2020 15:34:13 +00:00 pnp-tech
1583041 drwx 4096 Oct 22 2020 11:27:38 +00:00 EFI
16 -rw- 34 Oct 6 2020 13:56:03 +00:00 pnp-tech-time
17 -rw- 82790 Oct 6 2020 13:56:14 +00:00 pnp-tech-discovery-summary
18 -rw- 8425 Oct 6 2020 15:09:18 +00:00 1g_snake
19 -rw- 6858 Oct 7 2020 10:53:21 +00:00 100g_snake
20 -rw- 4705 Oct 22 2020 13:01:54 +00:00 startup-config

26975526912 bytes total (25538875392 bytes free)
Router# copy nvram:startup-config bootflash:
Destination filename [startup-config]?
3517 bytes copied in 0.647 secs (5436 bytes/sec)
```

Example 2: Copying a Startup Configuration File to USB Flash Disk

```
Router# dir usb0:
Directory of usb0:/
43261 -rwx 208904396 Oct 27 2020 14:10:20 -07:00
c8000aep-universalk9.17.02.01.SPA.bin
255497216 bytes total (40190464 bytes free)
Router# copy nvram:startup-config usb0:
Destination filename [startup-config]?
3172 bytes copied in 0.214 secs (14822 bytes/sec)
Router# dir usb0:
Directory of usb0:/
43261 -rwx 208904396 Oct 27 2020 14:10:20 -07:00
c8000aep-universalk9.17.02.01.SPA.bin
15:40:45 -07:00 startup-config255497216 bytes total (40186880 bytes free)
```

Example 3: Copying a Startup Configuration File to a TFTP Server

```
Router# copy bootflash:startup-config tftp:
Address or name of remote host []? 172.17.16.81
Destination filename [pe24_asr-1002-confg]? /auto/tftp-users/user/startup-config
!!
3517 bytes copied in 0.122 secs (28828 bytes/sec)
```

For more detailed information on managing configuration files, see the *Managing Configuration Files* section in the *Cisco IOS XE Configuration Fundamentals Configuration Guide*

Dynamic allocation of cores

Dynamic core allocations provide flexibility for users to leverage the CPU cores for different services and/or CEF/IPSec performances. The Cisco C8400 Series Secure Routers are equipped with either 24-cores or 16-cores and have the flexibility to allocate cores into the service plane from the data plane. The core allocation is based on the customer configuration of the different services available on these platforms.

You can use the **platform resource { service-plane-heavy | data-plane-heavy }** command to adjust the cores across service plane and data plane. However, you have to reboot the device for the configured profile to take effect.

```
Router(config)# platform resource { service-plane-heavy | data-plane-heavy }
```

Note By default, when a device boots up, the core allocation is data-plane-heavy for Autonomous mode and service-plane-heavy for Controller mode.

This command output shows the CPU cores allocation on C8475-G2:

```
Router# show platform software cpu allocation

CPU alloc information:

  Control plane cpu alloc: 0-1

  Data plane cpu alloc: 0,2-23

  Service plane cpu alloc: 0

  Slow control plane cpu alloc:
  Template used: default-data_plane_heavy
```

This command output shows the CPU cores allocation on C8455-G2:

```
Router# show platform software cpu allocation

CPU alloc information:

  Control plane cpu alloc: 0-1

  Data plane cpu alloc: 0,2-15

  Service plane cpu alloc: 0

  Slow control plane cpu alloc:
  Template used: default-data_plane_heavy
```

Filter the output of the show and more commands

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

To use this functionality, enter a **show** or **more** command followed by the “pipe” character (|); one of the keywords **begin**, **include**, or **exclude**; and a regular expression on which you want to search or filter (the expression is case sensitive):

show *command* | {**append** | **begin** | **exclude** | **include** | **redirect** | **section** | **tee**} *regular-expression*

The output matches certain lines of information in the configuration file. The following example illustrates how to use output modifiers with the **show interface** command when you want the output to include only lines in which the expression "protocol" appears:

```
Router# show interface | include protocol
FastEthernet0/0 is up, line protocol is up
Serial4/0 is up, line protocol is up
Serial4/1 is up, line protocol is up
Serial4/2 is administratively down, line protocol is down
Serial4/3 is administratively down, line protocol is down
```

Disable front-panel USB ports

SUMMARY STEPS

1. enable
2. configure terminal
3. platform usb disable
4. end
5. write memory

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** `Device# configure terminal`	Enters global configurationmode.
Step 3	platform usb disable **Example:** `Device # platform usb disable`	Disables USB ports. **Note** For re-enabling of front-panel usb ports, use the no form of command (**no platform usb disable**).
Step 4	end **Example:** `Device(config-router-af)# end`	Exits address family configuration mode and returns to privileged EXEC mode.
Step 5	write memory	Save to configuration.

	Command or Action	Purpose

		Example: Disabling Front-Panel USB Ports On Autonomous, Controller and vManage Mode
		The following example shows the configuration of disabling front-panel USB ports on autonomous, controller and vManage mode:
		`13RU#sh run | inc usb`
		`platform usb disable`
		`13RU#`
		To verify the disabling of USB ports on your device, use the following show command:
		show platform usb status
		`Router#show platform usb status`
		`USB enabled`
		`Router#`

Power off a router

Before you turn off a power supply, make certain the chassis is grounded and you perform a soft shutdown on the power supply. Not performing a soft shutdown will often not harm the router, but may cause problems in certain scenarios.

To perform a soft shutdown before powering off the router, enter the **reload** command to halt the system and then wait for ROM Monitor to execute before proceeding to the next step.

The following screenshot shows an example of this process:

```
Router# reload
Proceed with reload? [confirm]
...(Some messages are omitted here)
Initializing Hardware...
```

Place the power supply switch in the Off position after seeing this message.