



Chapter

4

Configuration and IOS Management Commands

THE CCNA EXAM TOPICS COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

- ✓ Use the setup feature on a Cisco router
- ✓ Log into a router in both user and privileged modes
- ✓ Find commands by using the help facilities
- ✓ Use commands on a router by using the editing command
- ✓ Set the router passwords, identification, and banners
- ✓ Configure an interface with IP addresses and subnet masks
- ✓ Copy the configuration to NVRAM



In this chapter, you will be introduced to the Cisco Internetwork Operating System (IOS). The IOS is what runs the Cisco routers and also some Cisco switches, which allows you to configure the devices as well.

You will learn how to configure a Cisco IOS router using both the initial setup mode and the Cisco IOS Command-Line Interface (CLI). Through the IOS interface, you can configure passwords, banners, and more. You will also learn the basics of router configurations in this chapter. The details covered in this chapter include the following:

- Understanding and configuring the Cisco Internetwork Operating System (IOS)
- Connecting to a router
- Bringing up a router
- Logging into a router
- Understanding the router prompts
- Understanding the CLI prompts
- Performing editing and help features
- Gathering basic routing information
- Setting router passwords
- Setting router banners
- Performing interface configurations
- Setting router hostnames

- Setting interface descriptions
- Viewing and saving router configurations
- Verifying routing configurations

It is important to have a firm understanding of the fundamentals taught in this chapter before you go on to the other chapters.

Cisco Router User Interface

The *Cisco Internetwork Operating System (IOS)* is the kernel of Cisco routers and most switches. Cisco has created what they call Cisco Fusion, which is supposed to make all Cisco devices run the same operating system. The reason they don't all run the same OS is because Cisco has acquired more devices than they have designed and built themselves. Almost all Cisco routers run the same IOS, but only about half of the switches currently run the Cisco IOS.

In this section, I'll give you a look at the Cisco IOS and how to configure a Cisco router step-by-step, first using setup mode and then through the Command-Line Interface (CLI).

Cisco Router IOS

The IOS was created to deliver network services and enable networked applications. The Cisco IOS runs on most Cisco routers and on some Cisco Catalyst switches, like the Catalyst 1900 switch (covered in Appendix B).

The Cisco router IOS software is used to complete the following on Cisco hardware:

- Carry network protocols and functions
- Connect high-speed traffic between devices
- Add security to control access and stop unauthorized network use
- Provide scalability for ease of network growth and redundancy
- Supply network reliability for connecting to network resources

You can access the Cisco IOS through the console port of a router, from a modem, or even through Telnet. Access to the IOS command line is called an EXEC session.

Connecting to a Cisco Router

You can connect to a Cisco router to configure the router, verify the configuration, and check statistics. There are different ways to connect to a Cisco router, but the first place you typically would connect to is the console port.

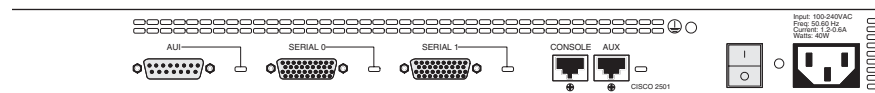
The *console port* is usually a RJ-45 connection on the back of the router. This is used to connect to and configure the router. No password is set on the console port by default.

Another way to connect to a Cisco router is through an *auxiliary port*. This is really the same as a console port and can be used as such. However, it also allows you to configure modem commands to allow a modem connection to the router. This means you can dial up a remote router and attach to the auxiliary port if the router is down and you need to configure it.

The third way to connect to a Cisco router is through the program *Telnet*. Telnet is an emulation program that emulates a dumb-terminal. You can then use Telnet to connect to any active interface on a router like an Ethernet or serial port.

Figure 4.1 shows an illustration of a 2501 Cisco router. Notice the different interfaces and connections.

FIGURE 4.1 2501 router



The 2501 router has two serial interfaces for WAN connection and one Attachment Unit Interface (AUI) connection for a 10Mbps Ethernet network connection. The 2501 router also has one console and one auxiliary connection via RJ-45 connectors.

Bringing Up a Router

When you first bring up a Cisco router, it will run a power-on self test (POST), and if that passes, it will look for and load the Cisco IOS from Flash memory if a file is present. Flash memory is an electronically erasable programmable read-only memory (EEPROM). The IOS will load and then look for a valid configuration called startup-config that is stored by default in nonvolatile RAM (NVRAM).

If there is no configuration in NVRAM, then the router will bring up what is called *setup mode*. This is a step-by-step process to help you configure a router. You can also enter setup mode at any time from the command line by typing the command `setup` from global configuration mode. Setup only covers some very global commands, but is helpful if you don't know how to configure certain protocols, like bridging or DECnet, for example.

Setup Mode

You actually have two options when using setup mode: *Basic Management* and *Extended Setup*. Basic Management only gives you enough configurations to allow connectivity to the router, whereas Extended Setup allows you to configure some global parameters as well as interface configuration parameters.

--- System Configuration Dialog ---

Would you like to enter the initial configuration dialog?
[yes/no]: **y**

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.

Basic Management Setup configures only enough connectivity for management of the system; Extended Setup will ask you to configure each interface on the system.

Would you like to enter basic management setup?
[yes/no]: **n**

First, would you like to see the current interface summary?
[yes]: **return**

Any interface listed with OK? value "NO" does not have a valid configuration

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	unassigned	NO	unset	up	up
FastEthernet0/1	unassigned	NO	unset	up	up

Configuring global parameters:

Enter host name [Router]: **Todd**

The enable secret is a password used to protect access to privileged EXEC and configuration modes. This password, after entered, becomes encrypted in the configuration.

Enter enable secret: **todd**

The enable password is used when you do not specify an enable secret password, with some older software versions, and some boot images.

Enter enable password: **todd**

% Please choose a password that is different from the enable secret

Enter enable password: **todd1**

Let's stop right here for a moment. Notice that setup mode asks you to configure two enable passwords. Passwords are covered later in this chapter, but you should understand that you really only use the enable secret password. The enable password is for pre-10.3 IOS routers. However, you must configure the password in setup mode, and it must be different. It will never be used if the enable secret is configured.

The next password is for setting up Telnet sessions to the router. The reason setup mode has you configure a Telnet (VTY) password is because if a password for the VTY lines is not set, you cannot by default telnet into a router.

The virtual terminal password is used to protect access to the router over a network interface.

Enter virtual terminal password: **todd**

```

Configure SNMP Network Management? [yes]:enter or no
  Community string [public]:enter
Configure DECnet? [no]:enter
Configure AppleTalk? [no]:enter
Configure IP? [yes]:enter
  Configure IGRP routing? [yes]: n
  Configure RIP routing? [no]:enter
Configure bridging? [no]:enter
Configure IPX? [no]:enter

```

The preceding commands can help you configure a protocol if you are not sure which commands you need to do so. However, using the Command-Line Interface (CLI) instead of setup mode gives you much more flexibility.

If you have an Async modem card installed in your router, you can have setup mode configure the modems for you.

Async lines accept incoming modems calls. If you will have users dialing in via modems, configure these lines.

```
Configure Async lines? [yes]: n
```

If your router has an ISDN BRI interface, you will be prompted for the ISDN switch type to be configured. Here is the router output:

```

BRI interface needs isdn switch-type to be configured
Valid switch types are:
[0] none.....Only if you don't want to configure
BRI.
[1] basic-1tr6....1TR6 switch type for Germany
[2] basic-5ess....AT&T 5ESS switch type for the US/Canada
[3] basic-dms100..Northern DMS-100 switch type for US/
Canada
[4] basic-net3....NET3 switch type for UK and Europe
[5] basic-ni.....National ISDN switch type
[6] basic-ts013...TS013 switch type for Australia
[7] ntt.....NTT switch type for Japan
[8] vn3.....VN3 and VN4 switch types for France
Choose ISDN BRI Switch Type [2]: 2

```

After the switch type is configured, you will be prompted for the interface configurations, including IP addresses.

The next section of the Extended Setup is configuring the interfaces. We only have two FastEthernet interfaces on this router, FastEthernet 0/0 and FastEthernet 0/1. The interfaces of a router will be discussed later in this chapter.

Configuring interface parameters:

```
Do you want to configure FastEthernet0/0 interface?
[yes]:return
  Use the 100 Base-TX (RJ-45) connector? [yes]:return
  Operate in full-duplex mode? [no]: y and return
  Configure IP on this interface? [yes]:return
  IP address for this interface: 1.1.1.1
  Subnet mask for this interface [255.0.0.0] :
255.255.0.0
  Class A network is 1.0.0.0, 16 subnet bits; mask is /
16
```

```
Do you want to configure FastEthernet0/1 interface?
[yes]:return
  Use the 100 Base-TX (RJ-45) connector? [yes]:return
  Operate in full-duplex mode? [no]:y and return
  Configure IP on this interface? [yes]:return
  IP address for this interface: 2.2.2.2
  Subnet mask for this interface [255.0.0.0] :
255.255.0.0
  Class A network is 2.0.0.0, 16 subnet bits; mask is /
16
```

This configuration is very basic, but it will allow you to get a router up and running quickly. Notice the mask is displayed as /16, which means 16 out of 32 bits are used. See Chapter 3 for more IP subnetting information.

The Extended Setup will now show the running configuration created:

The following configuration command script was created:

```
hostname Todd
```



```

enable secret 5 $1$B0wu$5F0m/EDdtRkQ4vy4a8qwC/
enable password todd1
line vty 0 4
password todd
snmp-server community public
!
no decnet routing
no appletalk routing
ip routing
no bridge 1
no ipx routing
!
interface FastEthernet0/0
media-type 100BaseX
full-duplex
ip address 1.1.1.1 255.255.0.0
no mop enabled
!
interface FastEthernet0/1
media-type 100BaseX
half-duplex
ip address 2.2.2.2 255.255.0.0
no mop enabled
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
!
end

```

[0] Go to the IOS command prompt without saving this config.

[1] Return back to the setup without saving this config.

[2] Save this configuration to nvram and exit.

Enter your selection [2]:0

The interesting part of the Extended Setup is the options you get at the end. You can go to CLI mode and discard the running-config [0]; you can go back

to setup to do it all over again [1]; or you can save this configuration to NVRAM, which is known as startup-config. This file would then be loaded every time the router is rebooted.

I chose “0” to go to the IOS and not save the file we created. This will take us to the CLI.

Command-Line Interface

The *Command-Line Interface (CLI)* is really the best way to configure a router because it gives you the most flexibility. To use the CLI, just say *no* to entering the Initial Configuration Dialog. After you say no, the router will come back with messages stating the status of all the router interfaces.

```
Would you like to enter the initial configuration dialog?
[yes]: n
```

```
Would you like to terminate autoinstall? [yes]:return
```

```
Press RETURN to get started!
```

```
00:00:42: %LINK-3-UPDOWN: Interface Ethernet0, changed
state to up
00:00:42: %LINK-3-UPDOWN: Interface Serial0, changed state
to down
00:00:42: %LINK-3-UPDOWN: Interface Serial1, changed state
to down
00:00:42: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Ethernet0, changed state to up
00:00:42: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0, changed state to down
00:00:42: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1, changed state to down
00:01:30: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Ethernet0, changed state to down
00:01:31: %LINK-5-CHANGED: Interface Serial0, changed
state to administrativelydown
00:01:31: %LINK-5-CHANGED: Interface Ethernet0, changed
state to administratively down
```

```
00:01:31: %LINK-5-CHANGED: Interface Serial1, changed
state to administratively down
00:01:32: %IP-5-WEBINST_KILL: Terminating DNS process
00:01:38: %SYS-5-RESTART: System restarted --
Cisco Internetwork Operating System Software
IOS (tm) 2500 Software (C2500-DS-L), Version 11.3(9),
RELEASE SOFTWARE (fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Tue 06-Apr-99 19:23 by dschwart
```

Logging into the Router

After the interface status messages appear and you press Return, the Router> prompt will appear. This is called user mode and is mostly used to view statistics, though it is also a stepping-stone to logging into privileged mode. You can only view and change the configuration of a Cisco router in privileged mode, which you enter with the command `enable`.

```
Router>
Router>enable
Router#
```

You now end up with a Router#, which indicates you are in privileged mode. You can both view and change the configuration in privileged mode. You can go back from privileged mode to user mode by using the `disable` command.

```
Router#disable
Router>
```

At this point you can type `logout` to exit the console.

```
Router>logout
```

```
Router con0 is now available
Press RETURN to get started.
```

Or you could just type `logout` or `exit` from the privileged mode prompt to log out.

```
Router>en
```

```
Router#logout
```

```
Router con0 is now available
Press RETURN to get started.
```

Overview of Router Modes

To configure from a CLI, you can make global changes to the router by typing **config terminal** (**config t** for short), which puts you in global configuration mode and changes what is known as the running-config. You can type **config** from the privileged mode prompt and then just press Return to take the default of terminal.

```
Router#config
```

```
Configuring from terminal, memory, or network
[terminal]?return
```

```
Enter configuration commands, one per line. End with
CNTL/Z.
```

```
Router(config)#
```

At this point you make changes that affect the router as a whole, hence the term global configuration mode.

To change the running-config, which is the current configuration running in Dynamic RAM (DRAM), you would use the command **config terminal**, or just **config t**. To change the configuration stored in NVRAM, which is known as startup-config, you would use the command **config memory**, or **config mem** for short. If you wanted to change a router configuration stored on a TFTP host (which is covered in Chapter 7), you would use the command **config network**, or **config net**.

However, understand that for a router to actually make a change to a configuration, it needs to put the configuration in RAM. So, if you actually type **config mem** or **config net**, you will replace the current running-config with the config stored in NVRAM or a configuration stored on a TFTP host.

CLI Prompts

It is important to understand the different prompts you can find when configuring a router so you know where you are at any time within configuration mode. In this section, we will demonstrate the prompts that are used on a Cisco router. Always check your prompts before making any changes to a router's configuration.

This section is not intended to show all the different commands offered but will, instead, describe the different prompts you will see throughout this chapter and the rest of the book.

Interfaces

To make changes to an interface, you use the `interface` command from global configuration mode:

```
Router(config)#interface ?
Async          Async interface
BVI           Bridge-Group Virtual Interface
Dialer        Dialer interface
FastEthernet  FastEthernet IEEE 802.3
Group-Async   Async Group interface
Lex           Lex interface
Loopback      Loopback interface
Multilink     Multilink-group interface
Null          Null interface
Port-channel  Ethernet Channel of interfaces
Tunnel        Tunnel interface
Virtual-Template Virtual Template interface
Virtual-TokenRing Virtual TokenRing
Router(config)#interface fastethernet 0/0
Router(config-if)#
```

Notice the prompt changed to `Router(config-if)#` to tell you that you are in interface configuration. It would be nice if it gave you an indication of what interface you were configuring, but it doesn't. This is probably one of the reasons Cisco administrators make more money than Windows administrators.

Subinterfaces

Subinterfaces allow you to create virtual interfaces within the router. The prompt then changes to `Router(config-subif)#`. (You can read more about subinterfaces in Chapter 10 and Appendix B.)

```
Router(config)#int f0/0.?
<0-4294967295> FastEthernet interface number
Router(config)#int f0/0.1
Router(config-subif)#
```

Line Commands

To configure user mode passwords, use the `line` command. The prompt then becomes `Router (config-line)#`.

```
Router#config t
```

```
Enter configuration commands, one per line. End with  
CNTL/Z.
```

```
Router(config)#line ?
```

```
<0-70> First Line number  
aux      Auxiliary line  
console Primary terminal line  
tty      Terminal controller  
vty      Virtual terminal
```

```
Router(config)#line console 0
```

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

The `line console 0` command is known as a major, or global, command, and any command typed from the `(config-line)` prompt is known as a subcommand.

Routing Protocol Configurations

To configure routing protocols like RIP and IGRP, use the prompt `(config-router)#`. (We cover routing protocols in Chapter 5.)

```
Router#config t
```

```
Enter configuration commands, one per line. End with  
CNTL/Z.
```

```
Router(config)#router rip
```

```
Router(config-router)#
```



It is not important that you understand what each of these commands does at this time. These will all be explained later in great detail. What you need to understand here is the different prompts available.

Editing and Help Features

You can use the Cisco advanced editing features to help you configure your router. By using a question mark (?) at any prompt, you can see the list of commands available from that prompt.

```
Router#?
Exec commands:
  access-enable   Create a temporary Access-List entry
  access-profile  Apply user-profile to interface
  access-template Create a temporary Access-List entry
  bfe             For manual emergency modes setting
  clear          Reset functions
  clock          Manage the system clock
  configure       Enter configuration mode
  connect        Open a terminal connection
  copy           Copy configuration or image data
  debug          Debugging functions (see also
'undebug')
  disable        Turn off privileged commands
  disconnect     Disconnect an existing network
connection
  enable         Turn on privileged commands
  erase          Erase flash or configuration memory
  exit           Exit from the EXEC
  help          Description of the interactive help
system
  lock          Lock the terminal
  login         Log in as a particular user
  logout        Exit from the EXEC
  mrinfo        Request neighbor and version information
from a multicast router

--More--
```

At this point, you can press the spacebar to get another page of information, or you can press Return to go one command at a time. You can also press any other key to quit and return to the prompt.

To find commands that start with a certain letter, use the letter and the question mark (?) with no space between them.

```
Router#c?  
clear clock configure connect copy
```

```
Router#c
```

Notice that by typing “c?”, we received a response of all the commands that start with “c”. Also notice that the Router# prompt appeared with our command still present. This is helpful when you have long commands and need the next possible command. Think about how tiresome it would be if you had to retype the entire command every time you used a question mark!

To find the next command in a string, type the first command and then a question mark.

```
Router#clock ?  
set Set the time and date
```

```
Router#clock set ?  
hh:mm:ss Current Time
```

```
Router#clock set 10:30:10 ?  
<1-31> Day of the month  
MONTH Month of the year  
Router#clock set 10:30:10 28 ?  
MONTH Month of the year
```

```
Router#clock set 10:30:10 28 may ?  
<1993-2035> Year
```

```
Router#clock set 10:30:10 28 may 2000 ?  
<cr>  
Router#
```

By typing the command clock, then a space and a question mark, you will get a list of the next possible commands and what they do. Notice that you should just keep typing a command, a space, and then a question mark until <cr> (carriage return) is your only option.

If you are typing commands and receive this:

```
Router#clock set 10:30:10
% Incomplete command.
```

then you know that the command string is not done. Just press the up arrow key to receive the last command entered, then continue with the command by using your question mark.

Also, if you receive this error:

```
Router(config)#access-list 110 permit host 1.1.1.1
                                     ^
% Invalid input detected at '^' marker.
```

notice that the ^ marks the point where you have entered the command incorrectly. This is very helpful.

If you receive this error:

```
Router#sh te
% Ambiguous command: "sh te"
```

it means you did not enter all the keywords or values required by this command. Use the question mark to find the command you need.

```
Router#sh te?
WORD  tech-support  terminal
```

Table 4.1 shows the list of enhanced editing commands available on a Cisco router.

TABLE 4.1 Enhanced Editing Commands

Command	Meaning
Ctrl+A	Moves your cursor to the beginning of the line
Ctrl+E	Moves your cursor to the end of the line
Esc+B	Moves back one word
Ctrl+F	Moves forward one character

TABLE 4.1 Enhanced Editing Commands (*continued*)

Command	Meaning
Esc+F	Moves forward one word
Ctrl+D	Deletes a single character
Backspace	Deletes a single character
Ctrl+R	Redisplays a line
Ctrl+U	Erases a line
Ctrl+W	Erases a word
Ctrl+Z	Ends configuration mode and returns to EXEC
Tab	Finishes typing a command for you

Another editing feature we need to mention is the automatic scrolling of long lines. In the following example, the command typed had reached the right margin and automatically moved ten spaces to the left. The dollar sign (\$) indicates that the line has been scrolled to the left.

```
Router#config t
Enter configuration commands, one per line. End with
CNTL/Z.
Router(config)#$ 110 permit host 171.10.10.10 0.0.0.0 host
```

You can review the router-command history with the commands shown in Table 4.2.

TABLE 4.2 Router-Command History

Command	Meaning
Ctrl+P or up arrow	Shows last command entered
Ctrl+N or down arrow	Shows previous commands entered

TABLE 4.2 Router-Command History (*continued*)

Command	Meaning
Show history	Shows last 10 commands entered by default
Show terminal	Shows terminal configurations and history buffer size
Terminal history size	Changes buffer size (max 256)

Here is an example of the `show history` command and how to change the history size, as well as how to verify it with the `show terminal` command.

Use the command `show history` to see the last 10 commands entered on the router.

```
Router#sh history
en
sh history
show terminal
sh cdp neig
sh ver
sh flash
sh int e0
sh history
sh int s0
sh int s1
```

We will now use the `show terminal` command to verify the terminal history size.

```
Router#sh terminal
Line 0, Location: "", Type: ""
[output cut]
History is enabled, history size is 10.
Full user help is disabled
Allowed transports are lat pad v120 telnet mop rlogin
nasi. Preferred is lat.
```

```
No output characters are padded
No special data dispatching characters
Group codes: 0
```

The command `terminal history size`, used from privileged mode, can change the size of the history buffer.

```
Router#terminal history size ?
<0-256> Size of history buffer
Router#terminal history size 25
```

Verify the change with the `show terminal` command.

```
Router#sh terminal
Line 0, Location: "", Type: ""
[output cut]
Editing is enabled.
History is enabled, history size is 25.
Full user help is disabled
Allowed transports are lat pad v120 telnet mop rlogin
nasi. Preferred is lat.
No output characters are padded
No special data dispatching characters
Group codes: 0
```

Gathering Basic Routing Information

The command `show version` will provide basic configuration for the system hardware as well as the software version, the names and sources of configuration files, and the boot images.

```
Router#sh version
Cisco Internetwork Operating System Software
IOS (tm) 2500 Software (C2500-JS-L), Version 12.0(8),
RELEASE SOFTWARE (fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Mon 29-Nov-99 14:52 by kpma
Image text-base: 0x03051C3C, data-base: 0x00001000
```

```
ROM: System Bootstrap, Version 11.0(10c), SOFTWARE  
BOOTFLASH: 3000 Bootstrap Software (IGS-BOOT-R), Version  
11.0(10c), RELEASE SOFTWARE (fc1)
```

```
RouterA uptime is 5 minutes  
System restarted by power-on  
System image file is "flash:c2500-js-1_120-8.bin"
```

```
cisco 2522 (68030) processor (revision N) with 14336K/  
2048K bytes of memory.  
Processor board ID 15662842, with hardware revision  
00000003  
Bridging software.  
X.25 software, Version 3.0.0.  
SuperLAT software (copyright 1990 by Meridian Technology  
Corp).  
TN3270 Emulation software.  
Basic Rate ISDN software, Version 1.1.  
1 Ethernet/IEEE 802.3 interface(s)  
2 Serial network interface(s)  
8 Low-speed serial(sync/async) network interface(s)  
1 ISDN Basic Rate interface(s)  
32K bytes of non-volatile configuration memory.  
16384K bytes of processor board System flash (Read ONLY)  
Configuration register is 0x2102
```

The `show version` command lets you know how long the router has been running, how it was restarted, the IOS filename running, the model hardware and processor versions, and the amount of DRAM. Also, the configuration register value is listed last. The configuration register is discussed in Chapter 7.

Setting the Passwords

There are five passwords used to secure your Cisco routers. The first two passwords are used to set your enable password, which is used to secure privileged mode. This will prompt a user for a password when the command `enable` is used. The other three are used to configure a password when user mode is accessed either through the console port, the auxiliary port, or Telnet.

Enable Passwords

You set the enable passwords from global configuration mode.

```
Router(config)#enable ?
  last-resort  Define enable action if no TACACS servers
               respond
  password     Assign the privileged level password
  secret       Assign the privileged level secret
  use-tacacs   Use TACACS to check enable passwords
```

Last-resort Is used if you set up authentication through a tacacs server and it is not available. This will allow the administrator to still enter the router. However, it is not used if the tacacs server is working.

Password Is used to set the enable password on older, pre-10.3 systems. Not used if an enable secret is set.

Secret Is the newer, encrypted password. Overrides the enable password if set.

Use-tacacs Tells the router to authenticate through a tacacs server. This is convenient if you have dozens or even hundreds of routers. How would you like to change the password on 200 routers? The tacacs server allows you to only have to change the password once.

```
Router(config)#enable secret todd
Router(config)#enable password todd
```

The enable password you have chosen is the same as your enable secret. This is not recommended. Re-enter the enable password.

If you try and set the enable secret and enable passwords to be the same, it will give you a nice, polite warning the first time, but if you type the same password again it will accept it. However, now neither password will work. If you don't have older legacy routers, don't bother to use the enable password.

User-mode passwords are assigned by using the `line` command.

```
Router(config)#line ?
  <0-4>      First Line number
  aux        Auxiliary line
```

```
console Primary terminal line
vty      Virtual terminal
```

Aux Is used to set the user-mode password for the auxiliary port. This is typically used for configuring a modem on the router but can be used as a console as well.

Console Is used to set a console user-mode password.

Vty Is used to set a Telnet password on the router. If the password is not set, then Telnet cannot be used by default.

To configure the user-mode passwords, you configure the line you want and use either the `login` or `no login` command to tell the router to prompt for authentication.

Auxiliary Password

To configure the auxiliary password, go to global configuration mode and type `line aux ?`. Notice that you only get a choice of 0-0 because there is only one port.

```
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#line aux ?
<0-0> First Line number
Router(config)#line aux 0
Router(config-line)#login
Router(config-line)#password todd
```

It is important to remember the `login` command, or the auxiliary port won't prompt for authentication.

Console Password

To set the console password, use the command `line console 0`. However, notice that when we tried to type `line console 0 ?` from the aux line configuration, we got an error. You can still type `line console 0` and it will accept it; however, the help screens do not work from that prompt. Type "exit" to get back one level.

```
Router(config-line)#line console ?
% Unrecognized command
```

```

Router(config-line)#exit
Router(config)#line console ?
<0-0> First Line number
Router(config)#line console 0
Router(config-line)#login
Router(config-line)#password todd1

```

Since there is only one console port, we can only choose line console 0.

Other Console Port Commands

There are a few other important commands to know for the console port.

The `exec-timeout 0 0` command sets the timeout for the console EXEC session to zero, or to never time out. To have fun with your friends at work, set it to `0 1`, which makes the console time out in 1 second! The way to fix that is to continually press the down arrow key while changing the timeout time with your free hand.

`logging synchronous` is a nice command, and it should be a default command, but it is not. What it does is stop console messages from popping up and disrupting input you are trying to type. This makes reading your input messages much easier.

Here is an example of how to configure both commands:

```

Router(config)#line con 0
Router(config-line)#exec-timeout ?
<0-35791> Timeout in minutes
Router(config-line)#exec-timeout 0 ?
<0-2147483> Timeout in seconds
<cr>
Router(config-line)#exec-timeout 0 0
Router(config-line)#logging synchronous

```

Telnet Password

To set the user-mode password for Telnet access into the router, use the `line vty` command. Routers that are not running the Enterprise edition of the Cisco IOS default to five VTY lines, 0 through 4. However, if you have the Enterprise edition, you will have significantly more. The router we are using

for this section has 198 (0–197). The best way to find out how many lines you have is to use the question mark.

```
Router(config-line)#line vty 0 ?
<1-197>Last Line Number
<cr>
Router(config-line)#line vty 0 197
Router(config-line)#login
Router(config-line)#password todd2
```

If you try to telnet into a router that does not have a VTY password set, you will receive an error stating that the connection is refused because the password is not set. You can tell the router to allow Telnet connections without a password by using the `no login` command.

```
Router(config-line)#line vty 0 197
Router(config-line)#no login
```

After your routers are configured with an IP address, you can use the Telnet program to configure and check your routers instead of having to use a console cable. You can use the Telnet program by typing `telnet` from any command prompt (DOS or Cisco). Telnet is covered in more detail in Chapter 7.

Encrypting Your Passwords

Only the enable secret password is encrypted by default. You need to manually configure the user-mode and enable passwords.

Notice that you can see all the passwords except the enable secret when performing a `show running-config` on a router.

```
Router#sh run
[output cut]
!
enable secret 5 $1$rFbM$8.aXocHg6yHrM/zzeNkAT.
enable password todd1
!
[output cut]
line con 0
password todd1
```

```

login
line aux 0
password todd
login
line vty 0 4
password todd2
login
line vty 5 197
password todd2
login
!
end

```

```
Router#
```

To manually encrypt your passwords, use the `service password-encryption` command. Here is an example of how to perform manual password encryption:

```

Router#config t
Enter configuration commands, one per line. End with
CNTL/Z.
Router(config)#service password-encryption
Router(config)#enable password todd
Router(config)#line vty 0 197
Router(config-line)#login
Router(config-line)#password todd2
Router(config-line)#line con 0
Router(config-line)#login
Router(config-line)#password todd1
Router(config-line)#line aux 0
Router(config-line)#login
Router(config-line)#password todd
Router(config-line)#exit
Router(config)#no service password-encryption
Router(config)#^Z

```

By typing the `show running-config` command, you can see the enable password and the line passwords are all encrypted.

```
Router#sh run
Building configuration...

[output cut]
!
enable secret 5 $1$rFbM$8.aXocHg6yHrM/zzeNkAT.
enable password 7 0835434A0D
!
[output cut]
!
line con 0
  password 7 111D160113
  login
line aux 0
  password 7 071B2E484A
  login
line vty 0 4
  password 7 0835434A0D
  login
line vty 5 197
  password 7 09463724B
  login
!
end

Router#
```

Banners

You can set a banner on a Cisco router so that when either a user logs into the router or an administrator telnets into the router, for example, a banner will give them the information you want them to have. Another reason for

having a banner is to add a security notice to users dialing into your inter-network. There are four different banners available:

```
Router(config)#banner ?
  LINE      c banner-text c, where 'c' is a delimiting
            character
  exec      Set EXEC process creation banner
  incoming  Set incoming terminal line banner
  login     Set login banner
  motd     Set Message of the Day banner
```

The Message of the Day is the most used and gives a message to every person dialing in or connecting to the router via Telnet, auxiliary port, or console port.

```
Router(config)#banner motd ?
  LINE c banner-text c, where 'c' is a delimiting
  character
Router(config)#banner motd #
Enter TEXT message. End with the character '#'.
$ized to be in Acme.com network, then you must disconnect
immediately.
#
Router(config)#^Z
Router#
00:25:12: %SYS-5-CONFIG_I: Configured from console by
console
Router#exit
```

```
Router con0 is now available
```

```
Press RETURN to get started.
```

```
If you are not authorized to be in Acme.com network, then
you must disconnect immediately.
```

```
Router>
```

The above MOTD banner tells anyone connecting to the router that they must either be authorized or they must disconnect. The part to understand is the

delimiting character. You can use any character you want, and it is used to tell the router when the message is done. So, you can't use the delimiting character in the message itself. One other thing to note is that once the message is complete, press Return, then the delimiting character, then Return. If you don't do that, it will still work, but if you have more than one banner, for example, it will combine them as one message and put them on one line.

These are the other banners:

Exec banner You can configure a line-activation (exec) banner to be displayed when an EXEC process (such as a line-activation or incoming connection to a VTY line) is created.

Incoming banner You can configure a banner to be displayed on terminals connected to reverse Telnet lines. This banner is useful for providing instructions to users who use reverse Telnet.

Login banner You can configure a login banner to be displayed on all connected terminals. This banner is displayed after the MOTD banner but before the login prompts. The login banner cannot be disabled on a per-line basis. To globally disable the login banner, you must delete the login banner with the `no banner login` command.

Router Interfaces

Interface configuration is one of the most important configurations of the router. Without interfaces, the router is useless. Interface configurations must be exact to communicate with other devices. Some of the configurations used to configure an interface are Network layer addresses, media-type, bandwidth, and other administrator commands.

Different routers use different methods to choose interfaces used on a router. For example, the following command shows a 2522 router with 10 serial interfaces, which are labeled 0 through 9:

```
Router(config)#int serial ?
<0-9> Serial interface number
```

At this point you must choose the interface you want to configure. Once you do that, you will be in interface configuration for that interface. The command to choose serial port 5, for example, would be

```
Router(config)#int serial 5
Router(config)-if)#
```

The 2522 router has one Ethernet 10BaseT port. Typing `interface ethernet 0` can configure the interface.

```
Router(config)#int ethernet ?
<0-0> Ethernet interface number
Router(config)#int ethernet 0
Router(config-if)#
```

The 2500 router, as previously demonstrated, is a fixed configuration router, which means that when you buy a model of router, you're stuck with that configuration. To configure an interface, you always use the `interface type number` sequence. However, the 2600, 3600, 4000, and 7000 series routers use a physical slot in the router and a port number on the module plugged into that slot. For example, on a 2600 router, the configuration would be `interface type slot/port`:

```
Router(config)#int fastethernet ?
<0-1> FastEthernet interface number
Router(config)#int fastethernet 0
% Incomplete command.
Router(config)#int fastethernet 0?
/
Router(config)#int fastethernet 0/?
<0-1> FastEthernet interface number
```

Notice that you cannot type `int fastethernet 0`. You must type the full command, which is `type slot/port`, or `int fastethernet 0/0`. You can type `int fa 0/0` as well.

To set the type of connector used, use the command `media-type`. However, this is typically auto-detected.

```
Router(config)#int fa 0/0
Router(config-if)#media-type ?
100BaseX Use RJ45 for -TX; SC F0 for -FX
MII      Use MII connector
```

Bringing Up an Interface

You can turn an interface off with the interface command `shutdown` or turn it on with the `no shutdown` command. If an interface is shut down, it will display administratively down when using the `show interface` command,

and the `show running-config` command will show the interface as shut down. All interfaces are shut down by default.

```
Router#sh int e0
Ethernet0 is administratively down, line protocol is down
[output cut]
```

Bring up an interface with the `no shutdown` command.

```
Router#config t
Enter configuration commands, one per line. End with
CNTL/Z.
Router(config)#int e0
Router(config-if)#no shutdown
Router(config-if)#^Z
00:57:08: %LINK-3-UPDOWN: Interface Ethernet0, changed
state to up
00:57:09: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Ethernet0, changed state to up
```

```
Router#sh int e0
Ethernet0 is up, line protocol is up
```

Configuring an IP Address on an Interface

You don't have to use IP on your routers; however, IP is typically used on all routers. To configure IP addresses on an interface, use the `ip address` command from interface configuration mode.

```
Router(config)#int e0
Router(config-if)#ip address 172.16.10.2 255.255.255.0
Router(config-if)#no shut
```

Don't forget to turn on an interface with the `no shut` command. Remember to look at the command `show interface e0`, for example, which will show you if it administratively shut down or not. `Show running-config` will also show you if the interface is shut down.

If you want to add a second subnet address to an interface, then you must use the `secondary` command. If you type another IP address and press Enter,

it will replace the existing IP address and mask. To add a secondary IP address, use the `secondary` command.

```
Router(config-if)#ip address 172.16.20.2 255.255.255.0
secondary
Router(config-if)#^Z
```

You can verify both addresses are configured on the interface with the `show running-config` command (`sh run` for short).

```
Router#sh run
Building configuration...
Current configuration:
[output cut]
!
interface Ethernet0
 ip address 172.16.20.2 255.255.255.0 secondary
 ip address 172.16.10.2 255.255.255.0
!
```

VIP Cards

If you have a 7000 or 7500 series router with VIP (Versatile Interface Processor) cards, you define an interface by using the `interface type slot/port adapter/port number`. For example

```
7000(config)#interface ethernet 2/0/0
```

Serial Interface Commands

To configure a serial interface, there are a couple of specifics that need to be discussed. Typically, the interface will be attached to a CSU/DSU type of device that provides clocking for the line. However, if you have a back-to-back configuration used in a lab environment, for example, one end must provide clocking. This would be the DCE end of the cable. Cisco routers, by default, are all DTE devices, and you must tell an interface to provide clocking if it is to act as a DCE device. You configure a DCE serial interface with the `clock rate` command.

```
Router#config t
Enter configuration commands, one per line. End with
CNTL/Z.
```



```

Router(config)#int s0
Router(config-if)#clock rate ?
    Speed (bits per second)
    1200
    2400
    4800
    9600
    19200
    38400
    56000
    64000
    72000
    125000
    148000
    250000
    500000
    800000
    1000000
    1300000
    2000000
    4000000

```

<300-4000000> Choose clockrate from list above

```

Router(config-if)#clock rate 64000
%Error: This command applies only to DCE interfaces
Router(config-if)#int s1
Router(config-if)#clock rate 64000

```

It does not hurt anything to try and put a clock rate on an interface. Notice that the `clock rate` command is in bits per second.

The next command you need to understand is the `bandwidth` command. Every Cisco router ships with a default serial link bandwidth of a T1, or 1.544Mbps. However, understand that this has nothing to do with how data is transferred over a link. The bandwidth of a serial link is used by routing protocols such as IGRP, EIGRP, and OSPF to calculate the best cost to a

remote network. If you are using RIP routing, then the bandwidth setting of a serial link is irrelevant.

```
Router(config-if)#bandwidth ?  
<1-10000000> Bandwidth in kilobits
```

```
Router(config-if)#bandwidth 64
```

Notice that unlike the `clock rate` command, the `bandwidth` command is configured in kilobits.

Hostnames

You can set the hostname of the router with the `hostname` command. This is only locally significant, which means it has no bearing on how the router performs name lookups on the internetwork.

```
Router#config t  
Enter configuration commands, one per line. End with  
CNTL/Z.  
Router(config)#hostname todd  
todd(config)#hostname Atlanta  
Atlanta(config)#
```

Even though it is tempting to configure the hostname after your own name, it is better served to name the router something significant to the location.

Descriptions

Setting descriptions on an interface is helpful to the administrator and, like the hostname, only locally significant. This is a helpful command because it can be used to keep track of circuit numbers, for example.

```
Atlanta(config)#int e0  
Atlanta(config-if)#description Sales Lan  
Atlanta(config-if)#int s0  
Atlanta(config-if)#desc Wan to Miami circuit:6fdda4321
```

You can view the description of an interface either with the `show running-config` command or the `show interface` command.

```
Atlanta#sh run
[cut]
interface Ethernet0
  description Sales Lan
  ip address 172.16.10.30 255.255.255.0
  no ip directed-broadcast
!
interface Serial0
  description Wan to Miami circuit:6fdda4321
  no ip address
  no ip directed-broadcast
  no ip mroute-cache
```

```
Atlanta#sh int e0
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0010.7be8.25db (bia
0010.7be8.25db)
  Description: Sales Lan
```

```
[cut]
Atlanta#sh int s0
Serial0 is up, line protocol is up
  Hardware is HD64570
  Description: Wan to Miami circuit:6fdda4321
[cut]
Atlanta#
```

Viewing and Saving Configurations

If you run through setup mode, it will ask you if you want to use the configuration you created. If you say yes, then it will copy the configuration running in DRAM, known as `running-config`, to NVRAM and name the file `startup-config`.

You can manually save the file from DRAM to NVRAM by using the `copy running-config startup-config` command. You can use the shortcut `copy run start` also.

```
Router#copy run start
Destination filename [startup-config]?return
Warning: Attempting to overwrite an NVRAM configuration
previously written by a different version of the system
image.
Overwrite the previous NVRAM configuration?[confirm]return
Building configuration...
```

Notice that the message stated we were trying to write over the older `startup-config`. The IOS had been just upgraded to version 12.8, and the last time the file was saved, 11.3 was running.

You can view the files by typing the command `show running-config` or `show startup-config` from privileged mode. The `sh run` command, which is the shortcut for `show running-config`, tells us that we are viewing the current configuration.

```
Router#sh run
Building configuration...
```

```
Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
ip subnet-zero
frame-relay switching
!
[cut]
```

The `sh start` command, which is the shortcut for the `show startup-config` command, shows us the configuration that will be used the next time

the router is reloaded and also shows us the amount of NVRAM used to store the startup-config file.

```
Router#sh start
Using 4850 out of 32762 bytes
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
!
ip subnet-zero
frame-relay switching
!
[cut]
```

You can delete the startup-config file by using the command `erase startup-config`. Once you perform this command, you will receive an error if you try to view the startup-config file.

```
Router#erase startup-config
Erasing the nvram filesystem will remove all files!
Continue? [confirm]
[OK]
Erase of nvram: complete
Router#sh start
%% Non-volatile configuration memory is not present
Router#
```

Verifying Your Configuration

Obviously, the `show running-config` would be the best way to verify your configuration, and the `show startup-config` would be the best way to verify the configuration used the next time the router is reloaded.

However, once you take a look at the running-config, and it appears that everything is in order, you can verify your configuration with utilities, like Ping and Telnet.

You can ping with different protocols, and you can see this by typing **ping ?** at the router user-mode or privileged mode prompt.

```
Router#ping ?
WORD      Ping destination address or hostname
appletalk Appletalk echo
decnet    DECnet echo
ip        IP echo
ipx       Novell/IPX echo
srb       srb echo
<cr>
```

To find a neighbor's Network layer address, you either need to go to the router or switch, or you can type **show cdp nei detail** to get the Network layer addresses, which you can use to ping with.

You can also use the trace program to find the path a packet takes as it traverses an internetwork. Trace can also be used with multiple protocols.

```
Router#trace ?
WORD      Trace route to destination address or
hostname
appletalk AppleTalk Trace
clns      ISO CLNS Trace
ip        IP Trace
oldvines  Vines Trace (Cisco)
vines     Vines Trace (Banyan)
<cr>
```

Telnet is the best tool, since it uses IP at the Network layer and TCP at the Transport layer to create a session with a remote host. If you can telnet into a device, your IP connectivity must be good. You can only telnet to IP addresses, and you can use Windows hosts or router prompts to telnet from.

```
Router#telnet ?
WORD      IP address or hostname of a remote system
<cr>
```

From the router prompt, you do not need to type the command `telnet`. If you just type a hostname or IP address, it will assume you want to telnet.

Verifying with the *Show Interface* Command

Another way to verify your configuration is by typing `show interface` commands. The first command is `show interface ?`, which shows us all the available interfaces to configure. The only interfaces that are not logical are Ethernet and Serial.

```
Router#sh int ?
Ethernet    IEEE 802.3
Null       Null interface
Serial     Serial
accounting Show interface accounting
crb       Show interface routing/bridging info
irb       Show interface routing/bridging info
<cr>
```

The next command is `show interface ethernet 0` and shows us the hardware address, logical address, and encapsulation method, as well as statistics on collisions.

```
Router#sh int e0
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0010.7b7f.c26c (bia
0010.7b7f.c26c)
  Internet address is 172.16.10.1/24
    MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec,
      reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10
sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:08:23, output 00:08:20, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
```

```
25 packets input, 2459 bytes, 0 no buffer
Received 25 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored,
0 abort
0 input packets with dribble condition detected
33 packets output, 7056 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
```

The most important status of the `show interface` command is the output of the line and data-link protocol status. If Ethernet 0 is up, line protocol is up, and the line is up and running.

```
RouterA#sh int e0
Ethernet0 is up, line protocol is up
```

The first parameter refers to the Physical layer and is up when it receives carrier detect. The second parameter refers to the Data Link layer and looks for keepalives from the connecting end.

```
RouterA#sh int s0
Serial0 is up, line protocol is down
```

If you see the line is up, but the protocol is down, you are having a clocking (keepalive) or framing issue. Check the keepalives on both ends to make sure they match; the clock rate is set, if needed; and the encapsulation type is the same on both ends.

```
RouterA#sh int s0
Serial0 is down, line protocol is down
```

If you see the line interface and protocol down, it is a cable or interface problem. Also, if one end is administratively shut down, then the remote end would show down and down. To turn on the interface, type the command `no shutdown` in interface configuration.

```
RouterB#sh int s0
Serial0 is administratively down, line protocol is down
```


The next command demonstrates the serial line and the Maximum Transmission Unit (MTU), which is 1500 bytes by default. It also shows the default bandwidth (BW) on all Cisco serial links: 1.544Kbs. This is used to determine the bandwidth of the line for routing protocols such as IGRP, EIGRP, and OSPF. Another important configuration to notice is the keep-alive, which is 10 seconds by default. Each router sends a keepalive message to its neighbor every 10 seconds. If both routers are not configured for the same keepalive time, it will not work.

You can clear the counters on the interface by typing the command `clear counters`.

```
Router#sh int s0
Serial0 is up, line protocol is up
  Hardware is HD64570
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10
sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
      Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored,
0 abort
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 16 interface resets
    0 output buffer failures, 0 output buffers swapped
out
    0 carrier transitions
      DCD=down DSR=down DTR=down RTS=down CTS=down
Router#clear counters ?
Ethernet IEEE 802.3
Null Null interface
Serial Serial
```

```

<cr>
Router#clear counters s0
Clear "show interface" counters on this interface
[confirm]return
Router#
00:17:35: %CLEAR-5-COUNTERS: Clear counter on interface
Serial0 by console
Router#

```

Using the *Show Controllers* Command

The `show controllers` command displays information about the physical interface itself. It will also give you the type of serial cable plugged into a serial port. Typically this will only be a DTE cable, which then plugs into a type of Data Service Unit (DSU).

```

Router#sh controllers s 0
HD unit 0, idb = 0x1229E4, driver structure at 0x127E70
buffer size 1524 HD unit 0, V.35 DTE cable
cpb = 0xE2, eda = 0x4140, cda = 0x4000

Router#sh controllers s 1
HD unit 1, idb = 0x12C174, driver structure at 0x131600
buffer size 1524 HD unit 1, V.35 DCE cable
cpb = 0xE3, eda = 0x2940, cda = 0x2800

```

Notice that serial 0 has a DTE cable, whereas the serial 1 connection has a DCE cable. Serial 1 would have to provide clocking with the `clock rate` command. Serial 0 would get its clocking from the DSU. Also, understand that this is the only command that needs to have a space after the command `serial`.

```

Router#sh controllers s1
                        ^
% Invalid input detected at '^' marker.

```

Summary

In this chapter, we introduced you to the Cisco Internetwork Operating System (IOS). It is important that you have a firm understanding of the basics offered in this chapter before you move on to the other chapters in this book. The basics of Cisco routers covered in this chapter include the following:

- Understanding the Cisco Internetwork Operating System (IOS) and how you can use the IOS to run and configure Cisco routers
- Connecting to a router with console connections and LAN connections
- Bringing up a router and entering setup mode
- Logging into a router and understanding the difference between user mode and privileged mode
- Understanding router prompts within router configuration mode
- Understanding editing and help features available from the router CLI
- Gathering basic routing information using the `show` commands
- Setting router passwords for both usermode and privileged mode access
- Setting router banners for identification
- Performing interface configurations to set the IP address on an interface
- Setting router hostnames for router identification
- Setting interface descriptions to identify each interface on a router
- Viewing and saving router configurations using the `show` commands and the `copy run start` command
- Verifying routing configurations using `show` commands