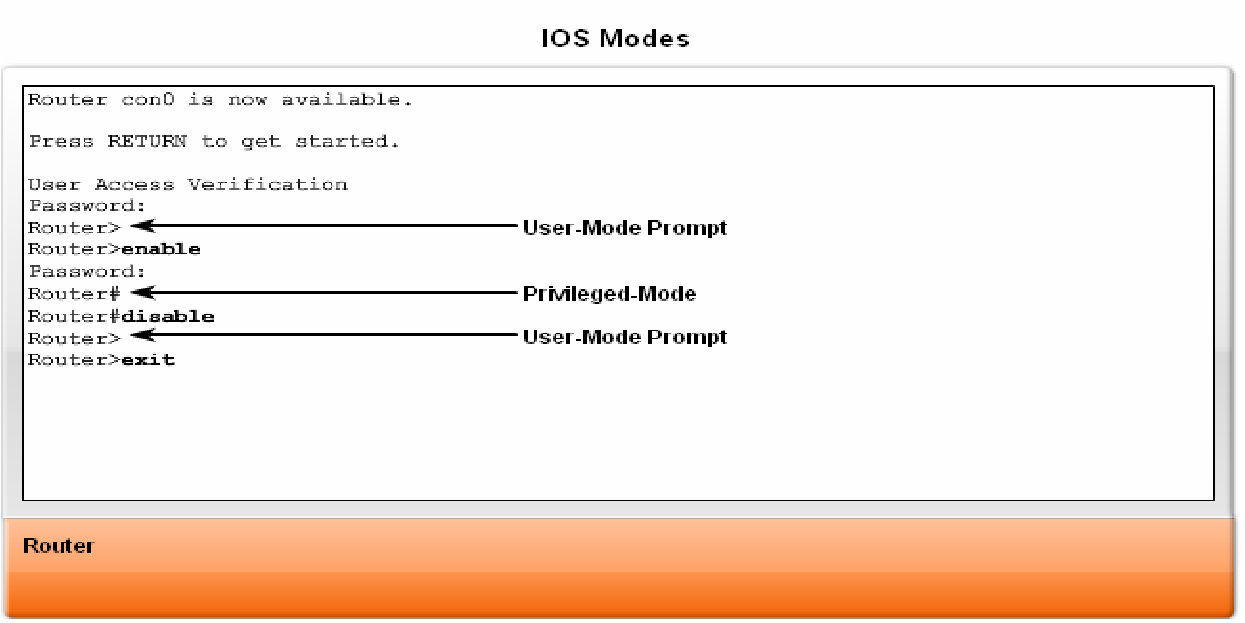
Router#disable

Router>

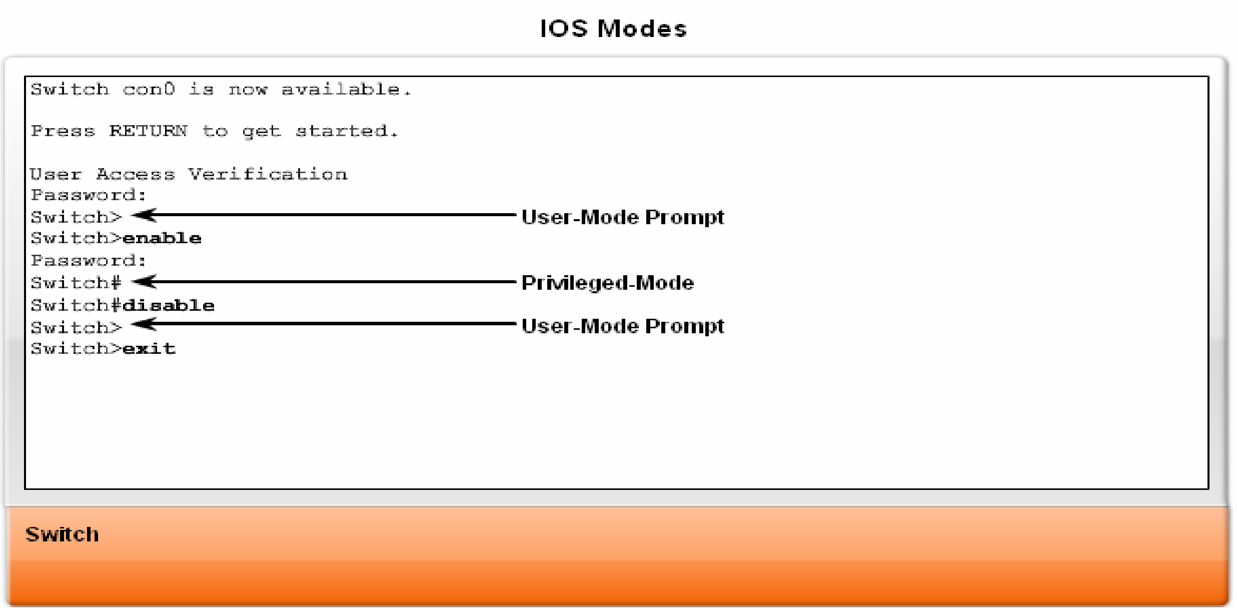


IOS Modes

```
Router con0 is now available.

Press RETURN to get started.

User Access Verification
Password:
Router>
Router>enable
Password:
Router#
Router#disable
Router>
Router>exit
```



IOS Modes

```
Switch con0 is now available.

Press RETURN to get started.

User Access Verification
Password:
Switch>
Switch>enable
Password:
Switch#
Switch#disable
Switch>
Switch>exit
```

Basic IOS Command Structure

Each IOS command has specific format or syntax and is executed at the appropriate prompt. The general syntax for a command is the command followed by any appropriate keywords and arguments. Some commands include a subset of keywords and arguments that provide additional functionality. The figure shows these parts of a command.

The command is the initial word or words entered in the command line. The commands are not case-sensitive. Following the command are one or more keywords and arguments.

The keywords describe specific parameters to the command interpreter. For example, the show command is used to display information about the device. This command has various keywords that can be used to define what particular output should be displayed. For example:

Switch#show running-config

The command show is followed by the keyword running-config. The keyword specifies that the running configuration is to be displayed as the output.

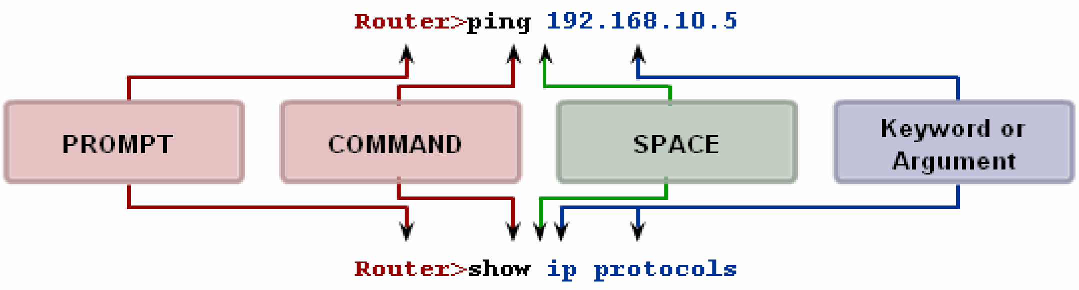
A command might require one or more arguments. Unlike a keyword, an argument is generally not a predefined word. An argument is a value or variable defined by the user. As an example, when applying a description to an interface with the description command, enter a line such as this:

Switch(config-if)#description MainHQ Office Switch

The command is: description. The argument is: MainHQ Office Switch. The user defines the argument. For this command, the argument can be any text string of up to 80 characters.

After entering each complete command, including any keywords and arguments, press the <Enter> key to submit the command to the command interpreter.

Basic IOS Command Structure



```
Router>ping 192.168.10.5
```

```
Router>show ip protocols
```

Prompt commands are followed by a space and then the keyword or arguments.

IOS Conventions

The figure and the following examples demonstrate some conventions for documenting IOS commands.

For the ping command:

Format:

Router>ping IP address

Example with values:

Router>ping 10.10.10.5

The command is ping and the argument is the IP address.

Similarly, the syntax for entering the traceroute command is:

Format:

Switch>traceroute IP address

Example with values:

Switch>traceroute 192.168.254.254

The command is traceroute and the argument is the IP address or web name.

Commands are used to execute an action, and the keywords are used to identify where or how to execute the command.

For another example, return to examining the description command.

Format:

Router(config-if)#description string

Example with values:

Switch(config-if)#description Interface to Building a LAN

The command is description , and the argument applied to the interface is the text string, Interface to Building a LAN. Once the command is executed, that description will be applied to the particular interface.

Using CLI Help

The IOS has several forms of help available:

- Context-sensitive help
- Command Syntax Check
- Hot Keys and Shortcuts

Context-Sensitive Help

The context-sensitive help provides a list of commands and the arguments associated with those commands within the context of the current mode. To access context-sensitive help, enter a question mark, ?, at any prompt. There is an immediate response without the need to use the <Enter> key.

One use of context-sensitive help is to get a list of available commands. This can be used when you are unsure of the name for a command or you want to see if the IOS supports a particular command in a particular mode.

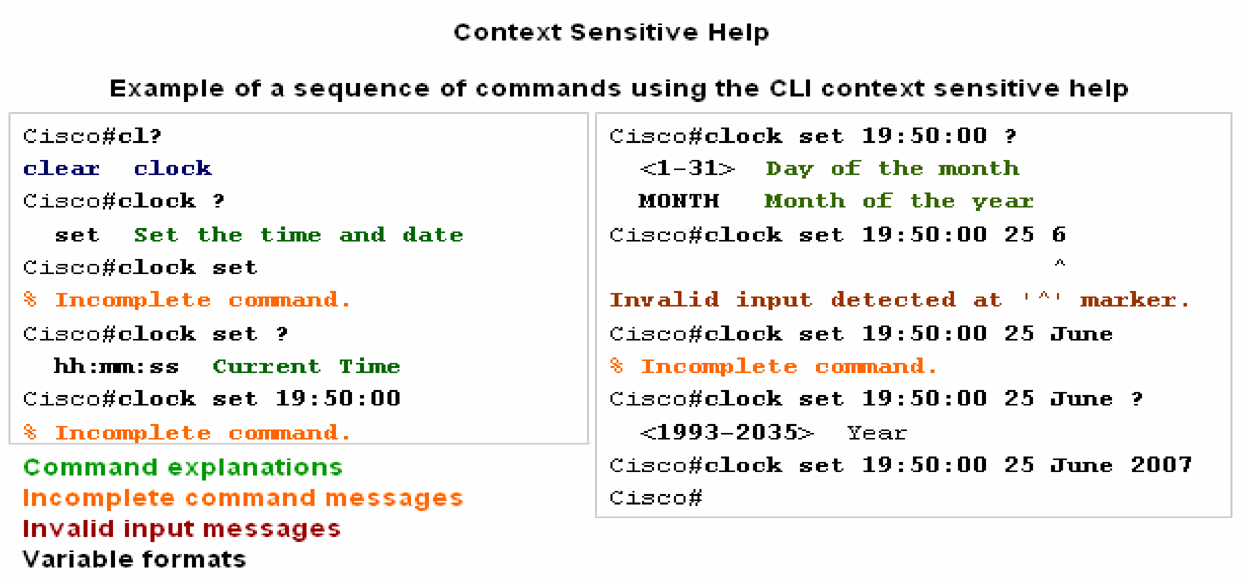
For example, to list the commands available at the user EXEC level, type a question mark ? at the Router> prompt.

Another use of context-sensitive help is to display a list of commands or keywords that start with a specific character or characters. After entering a character sequence, if a question mark is immediately entered-without a space-the IOS will display a list of commands or keywords for this context that start with the characters that were entered.

For example, enter sh? to get a list of commands that begin with the character sequence sh.

A final type of context-sensitive help is used to determine which options, keywords, or arguments are matched with a specific command. When entering a command, enter a space followed by a ? to determine what can or should be entered next.

As shown in the figure, after entering the command clock set 19:50:00, we can enter the ? to determine the options or keywords that fit with this command.



Context Sensitive Help

Example of a sequence of commands using the CLI context sensitive help

```
Cisco#cl?
clear   clock
Cisco#clock ?
  set   Set the time and date
Cisco#clock set
% Incomplete command.
Cisco#clock set ?
  hh:mm:ss   Current Time
Cisco#clock set 19:50:00
% Incomplete command.
```

```
Cisco#clock set 19:50:00 ?
  <1-31>   Day of the month
  MONTH    Month of the year
Cisco#clock set 19:50:00 25 6
                             ^
Invalid input detected at '^' marker.
Cisco#clock set 19:50:00 25 June
% Incomplete command.
Cisco#clock set 19:50:00 25 June ?
  <1993-2035>   Year
Cisco#clock set 19:50:00 25 June 2007
Cisco#
```

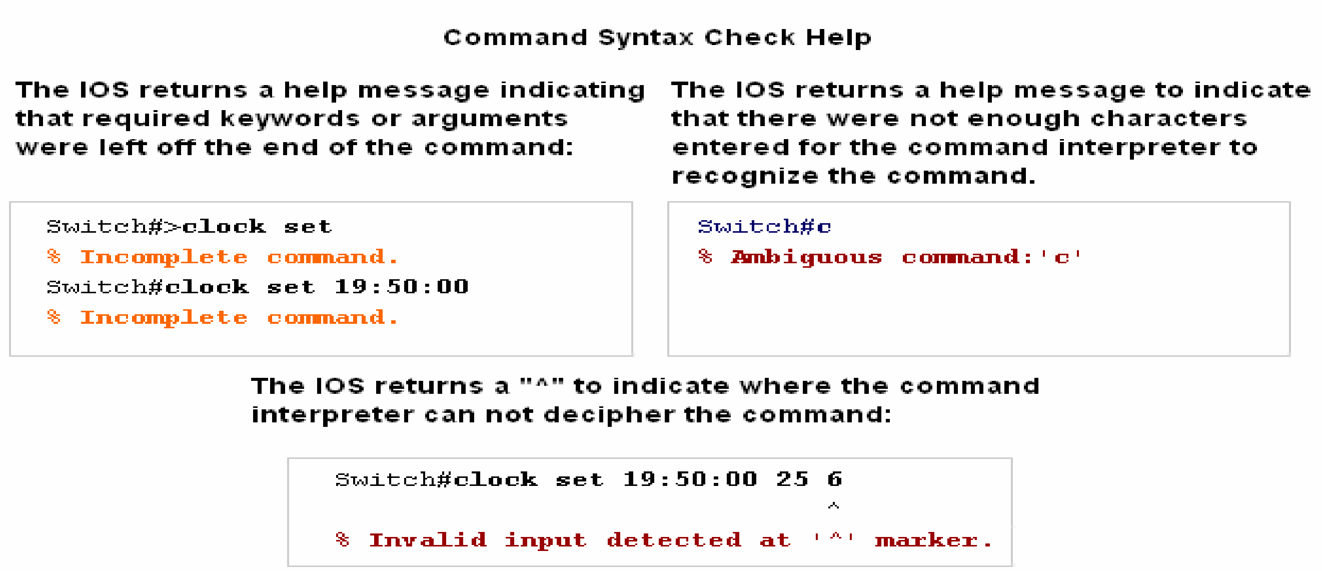
Command Syntax Check

When a command is submitted by pressing the <Enter> key, the command line interpreter parses the command from left to right to determine what action is being requested. The IOS generally only provides negative feedback. If the interpreter understands the command, the requested action is executed and the CLI returns to the appropriate prompt. However, if the interpreter cannot understand the command being entered, it will provide feedback describing what is wrong with the command.

There are three different types of error messages:

- Ambiguous command
- Incomplete command
- Incorrect command

See the figure for the types of errors and the remedies.



Command Syntax Check Help

The IOS returns a help message indicating that required keywords or arguments were left off the end of the command:

```
Switch#>clock set
% Incomplete command.
Switch#clock set 19:50:00
% Incomplete command.
```

The IOS returns a help message to indicate that there were not enough characters entered for the command interpreter to recognize the command.

```
Switch#c
% Ambiguous command:'c'
```

The IOS returns a "^" to indicate where the command interpreter can not decipher the command:

```
Switch#clock set 19:50:00 25 6
                              ^
% Invalid input detected at '^' marker.
```

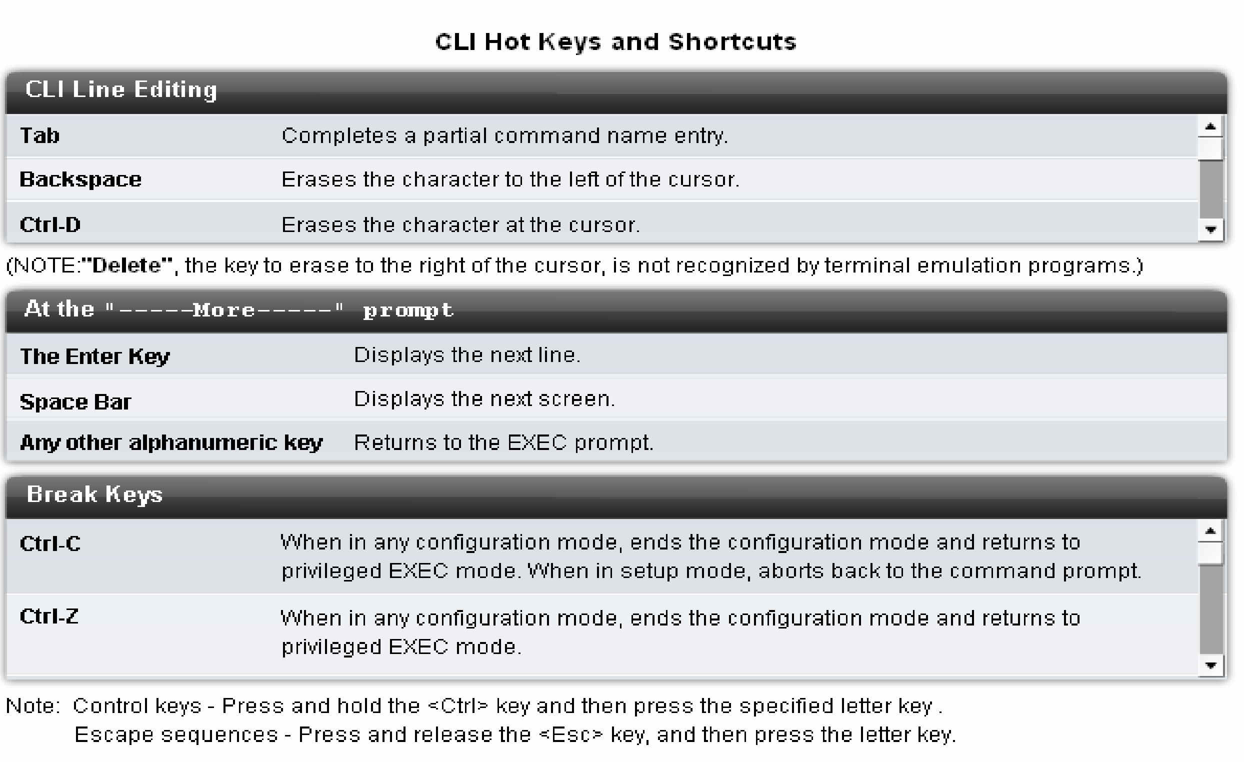
Hot Keys and Shortcuts

The IOS CLI provides hot keys and shortcuts that make configuring, monitoring, and troubleshooting easier.

The figure shows most of the shortcuts. The following are worthy of special note:

- Tab - Completes the remainder of the command or keyword
- Ctrl-R - Redisplays a line
- Ctrl-Z - Exits configuration mode and returns to the EXEC
- Down Arrow - Allows user to scroll forward through former commands
- Up Arrow - Allows user to scroll backward through former commands

- Ctrl-Shift-6 - Allows the user to interrupt an IOS process such as ping or traceroute



CLI Hot Keys and Shortcuts

CLI Line Editing	
Tab	Completes a partial command name entry.
Backspace	Erases the character to the left of the cursor.
Ctrl-D	Erases the character at the cursor.

(NOTE:**"Delete"**, the key to erase to the right of the cursor, is not recognized by terminal emulation programs.)

At the "-----More-----" prompt	
The Enter Key	Displays the next line.
Space Bar	Displays the next screen.
Any other alphanumeric key	Returns to the EXEC prompt.

Break Keys	
Ctrl-C	When in any configuration mode, ends the configuration mode and returns to privileged EXEC mode. When in setup mode, aborts back to the command prompt.
Ctrl-Z	When in any configuration mode, ends the configuration mode and returns to privileged EXEC mode.

Note: Control keys - Press and hold the <Ctrl> key and then press the specified letter key .
Escape sequences - Press and release the <Esc> key, and then press the letter key.

show version

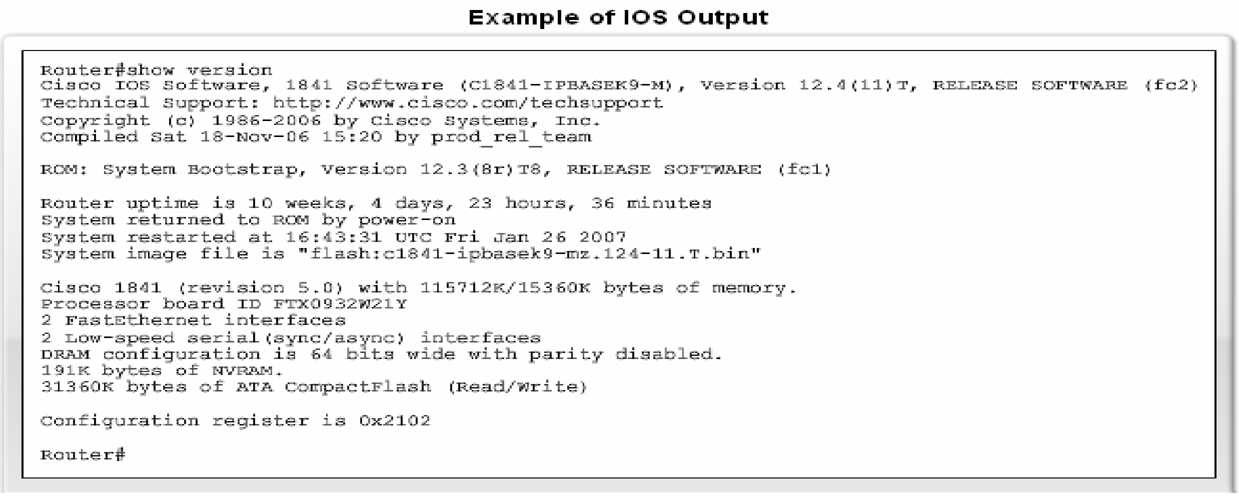
Displays information about the currently loaded software version, along with hardware and device information. Some of the information shown from this command are:

- Software Version - IOS software version (stored in flash)
- Bootstrap Version - Bootstrap version (stored in Boot ROM)
- System up-time - Time since last reboot
- System restart info - Method of restart (e.g., power cycle, crash)
- Software image name - IOS filename stored in flash
- Router Type and Processor type - Model number and processor type
- Memory type and allocation (Shared/Main) - Main Processor RAM and Shared Packet I/O buffering
- Software Features - Supported protocols / feature sets
- Hardware Interfaces - Interfaces available on router
- Configuration Register - Sets bootup specifications, console speed setting, and related parameters.

The figure shows a sample of typical show version output.

- **show arp** - Displays the ARP table of the device.

- **show mac**-address-table - (switch only) Displays the MAC table of a switch.
- **show startup-config** - Displays the saved configuration located in NVRAM.
- **show running-config** - Displays the contents of the currently running configuration file or the configuration for a specific interface, or map class information.
- **show ip interfaces** - Displays IPv4 statistics for all interfaces on a router. To view the statistics for a specific interface, enter the show ip interfaces command followed by the specific interface slot/port number. Another important format of this command is show ip interface brief. This is useful to get a quick summary of the interfaces and their operational state.



Example of IOS Output

```
Router#show version
Cisco IOS Software, 1841 Software (C1841-IPBASEK9-M), Version 12.4(11)T, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Sat 18-Nov-06 15:20 by prod_rel_team

ROM: System Bootstrap, Version 12.3(8r)T8, RELEASE SOFTWARE (fc1)

Router uptime is 10 weeks, 4 days, 23 hours, 36 minutes
System returned to ROM by power-on
System restarted at 16:43:31 UTC Fri Jan 26 2007
System image file is "flash:c1841-ipbasek9-mz.124-11.T.bin"

Cisco 1841 (revision 5.0) with 115712K/15360K bytes of memory.
Processor board ID FTX0932W21Y
2 FastEthernet interfaces
2 Low-speed serial(sync/async) interfaces
DRAM configuration is 64 bits wide with parity disabled.
191K bytes of NVRAM.
31360K bytes of ATA CompactFlash (Read/Write)

Configuration register is 0x2102

Router#
```

Configuring Passwords

The passwords introduced here are:

Console password - limits device access using the console connection

Enable password - limits access to the privileged EXEC mode

Enable secret password - encrypted, limits access to the privileged EXEC mode

VTY password - limits device access using Telnet

Console Password

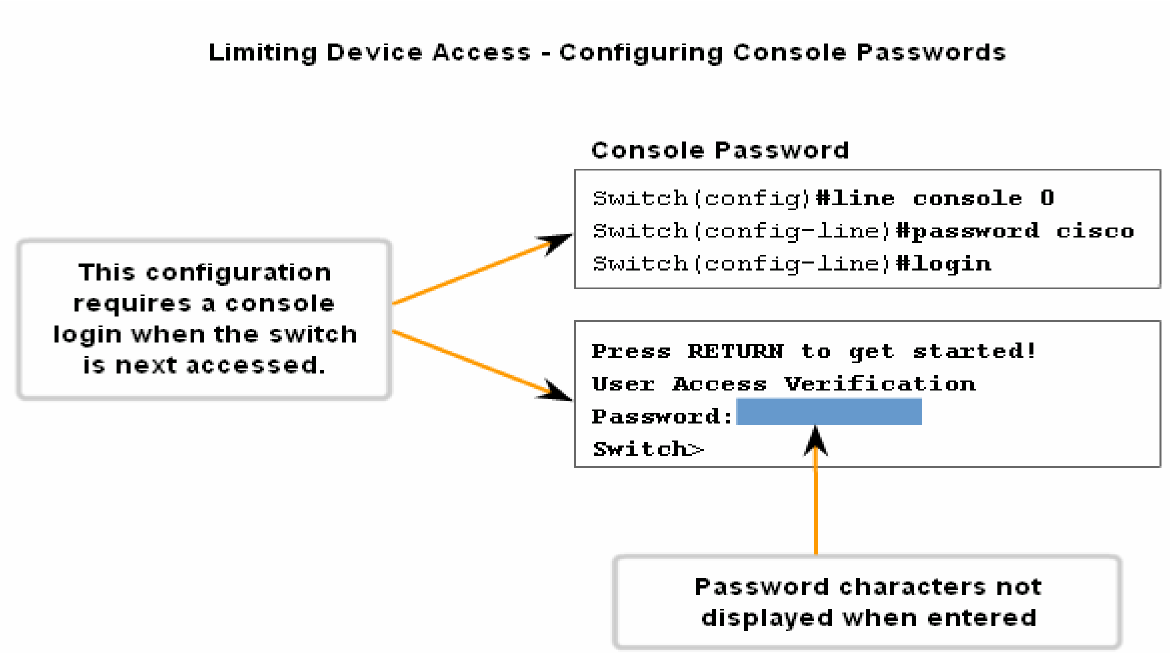
The console port of a Cisco IOS device has special privileges. The console port of network devices must be secured, at a bare minimum, by requiring the user to supply a strong password. This reduces the chance of unauthorized personnel physically plugging a cable into the device and gaining device access.

The following commands are used in global configuration mode to set a password for the console line:

```
Switch(config)#line console 0
Switch(config-line)#password password
```

Switch(config-line)#login

From global configuration mode, the command line console 0 is used to enter line configuration mode for the console. The z`x ero is used to represent the first (and in most cases only) console interface for a router.
The second command, password password specifies a password on a line.
The login command configures the router to require authentication upon login. When login is enabled and a password set, there will be a prompt to enter a password.



Limiting Device Access - Configuring Console Passwords

Console Password

```
Switch(config)#line console 0
Switch(config-line)#password cisco
Switch(config-line)#login
```

This configuration requires a console login when the switch is next accessed.

```
Press RETURN to get started!
User Access Verification
Password:
Switch>
```

Password characters not displayed when entered

Enable and Enable Secret Passwords

To provide additional security, use the enable password command or the enable secret command. Either of these commands can be used to establish authentication before accessing privileged EXEC (enable) mode.
Always use the enable secret command, not the older enable password command, if possible. The enable secret command provides greater security because the password is encrypted. The enable password command can be used only if enable secret has not yet been set.

The enable password command would be used if the device uses an older copy of the Cisco IOS software that does not recognize the enable secret command.
The following commands are used to set the passwords:
Router(config)#enable passwordpassword
Router(config)#enable secret password

Note: If no enable password or enable secret password is set, the IOS prevents privileged EXEC access from a Telnet session.

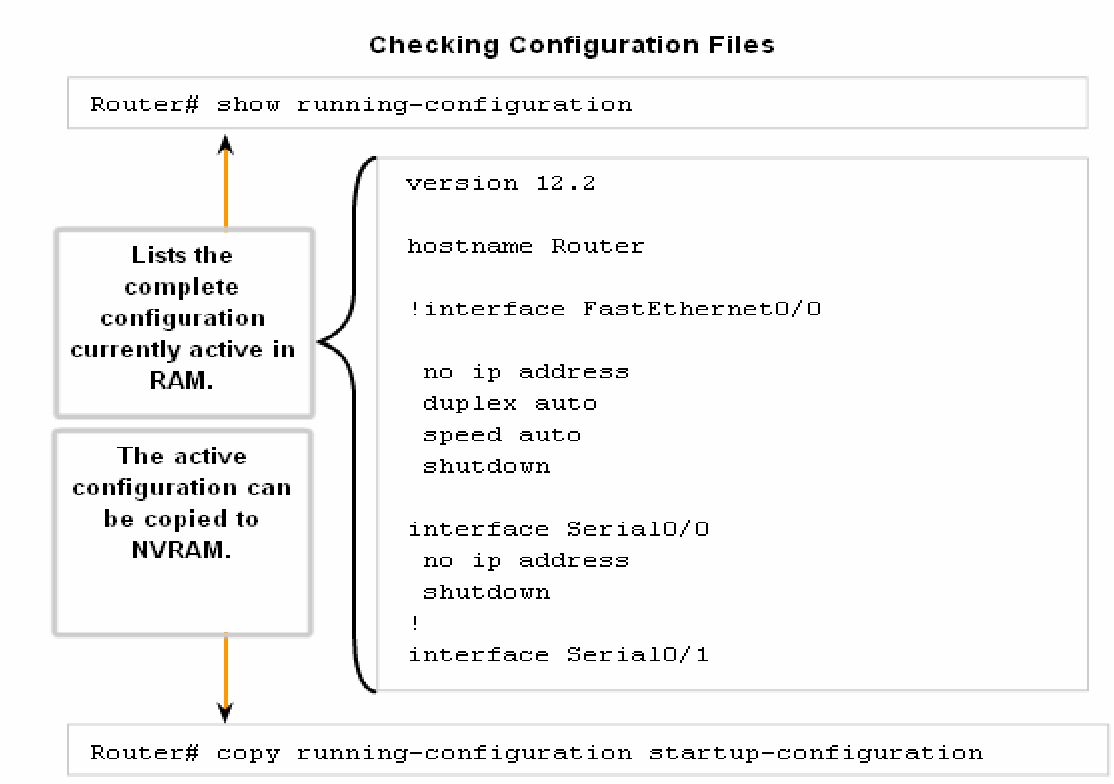
VTY Password

The vty lines allow access to a router via Telnet. By default, many Cisco devices support five VTY lines that are numbered 0 to 4. A password needs to be set for all available vty lines. The same password can be set for all connections. However, it is often desirable that a unique password be set for one line to provide a fall-back for administrative entry to the device if the other connections are in use.

The following commands are used to set a password on vty lines:

- Router(config)#line vty 0 4
- Router(config-line)#password password
- Router(config-line)#login

By default, the IOS includes the login command on the VTY lines. This prevents Telnet access to the device without first requiring authentication. If, by mistake, the no login command is set, which removes the requirement for authentication, unauthorized persons could connect to the line using Telnet. This would be a major security risk.



Checking Configuration Files

```
Router# show running-configuration
```

```
version 12.2

hostname Router

!interface FastEthernet0/0

 no ip address
 duplex auto
 speed auto
 shutdown

interface Serial0/0
 no ip address
 shutdown
!
interface Serial0/1
```

```
Router# copy running-configuration startup-configuration
```

Lists the complete configuration currently active in RAM.
The active configuration can be copied to NVRAM.

Backing Up Configurations Offline

Configuration files should be stored as backup files in the event of a problem. Configuration files can be stored on a Trivial File Transfer Protocol (TFTP) server, a CD, a USB memory stick, or a floppy disk stored in a safe place. A configuration file should also be included in the network documentation.

Backup Configuration on TFTP Server

As shown in the figure, one option is to save the running configuration or the startup configuration to a TFTP server. Use either the copy running-config tftp or copy startup-config tftp command and follow these steps:

1. Enter the copy running-config tftp command.
2. Enter the IP address of the host where the configuration file will be stored.
3. Enter the name to assign to the configuration file.
4. Answer yes to confirm each choice.

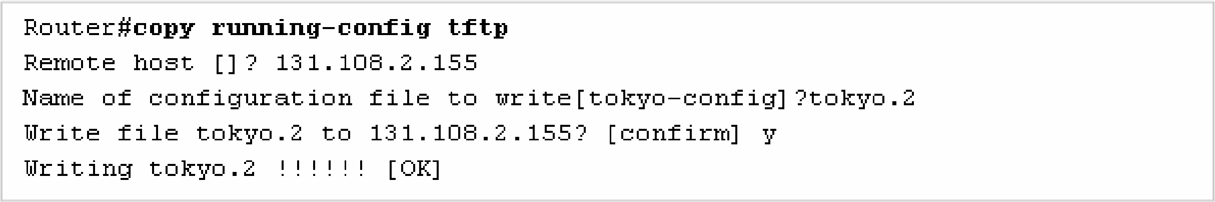
See the figure to view this process.

Removing All Configurations

If undesired changes are saved to the startup configuration, it may be necessary to clear all the configurations. This requires erasing the startup configuration and restarting the device.

The startup configuration is removed by using the erase startup-config command.

To erase the startup configuration file use erase NVRAM:startup-config or erase startup-configat the privileged EXEC mode prompt:
Router#erase startup-config.



```
Router#copy running-config tftp
Remote host []? 131.108.2.155
Name of configuration file to write[tokyo-config]?tokyo.2
Write file tokyo.2 to 131.108.2.155? [confirm] y
Writing tokyo.2 !!!!!! [OK]
```