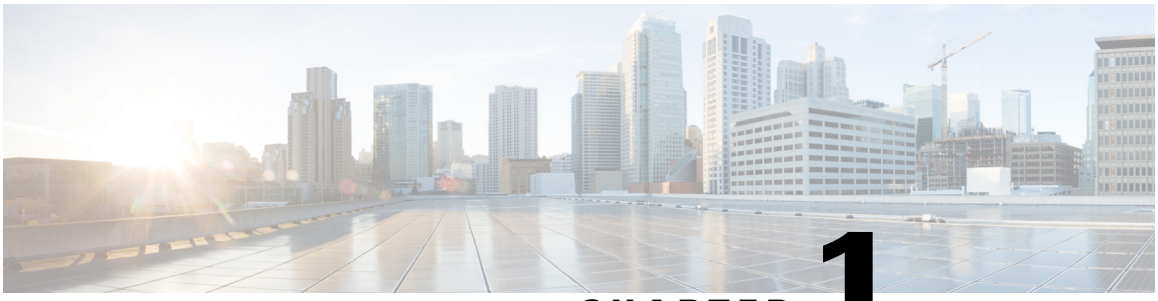
Configuration Fundamentals Configuration Guide, Cisco IOS Release 15SY

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

Using the Cisco IOS Command-Line Interface

The Cisco IOS command-line interface (CLI) is the primary user interface used for configuring, monitoring, and maintaining Cisco devices. This user interface allows you to directly and simply execute Cisco IOS commands, whether using a router console or terminal, or using remote access methods.

This chapter describes the basic features of the Cisco IOS CLI and how to use them. Topics covered include an introduction to Cisco IOS command modes, navigation and editing features, help features, and command history features.

Additional user interfaces include Setup mode (used for first-time startup), the Cisco Web Browser, and user menus configured by a system administrator. For information about Setup mode, see Using Setup Mode to Configure a Cisco Networking Device and Using AutoInstall to Remotely Configure Cisco Networking Devices. For information on issuing commands using the Cisco Web Browser, see "Using the Cisco Web Browser User Interface". For information on user menus, see "Managing Connections, Menus, and System Banners".

For a complete description of the user interface commands in this chapter, see the *Cisco IOS Configuration Fundamentals Command Reference*. To locate documentation of other commands that appear in this chapter, use the Cisco IOS Master Command List, All Releases .

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Cisco IOS XE CLI Command Modes Overview

To aid in the configuration of Cisco devices, the Cisco IOS XE command-line interface is divided into different command modes. Each command mode has its own set of commands available for the configuration, maintenance, and monitoring of router and network operations. The commands available to you at any given time depend on the mode you are in. Entering a question mark (**?**) at the system prompt (router prompt) allows you to obtain a list of commands available for each command mode.

The use of specific commands allows you to navigate from one command mode to another. The standard order that a user would access the modes is as follows: user EXEC mode; privileged EXEC mode; global configuration mode; specific configuration modes; configuration submodes; and configuration subsubmodes.

When you start a session on a router, you generally begin in *user EXEC mode*, which is one of two access levels of the EXEC mode. For security purposes, only a limited subset of EXEC commands are available in user EXEC mode. This level of access is reserved for tasks that do not change the configuration of the router, such as determining the router status.

In order to have access to all commands, you must enter *privileged EXEC mode*, which is the second level of access for the EXEC mode. Normally, you must enter a password to enter privileged EXEC mode. In privileged EXEC mode, you can enter any EXEC command, because privileged EXEC mode is a superset of the user EXEC mode commands.

Most EXEC mode commands are one-time commands, such as **show** or **more** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. EXEC mode commands are not saved across reboots of the router.

From privileged EXEC mode, you can enter *global configuration mode*. In this mode, you can enter commands that configure general system characteristics. You also can use global configuration mode to enter specific configuration modes. Configuration modes, including global configuration mode, allow you to make changes to the running configuration. If you later save the configuration, these commands are stored across router reboots.

From global configuration mode you can enter a variety of protocol-specific or feature-specific configuration modes. The CLI hierarchy requires that you enter these specific configuration modes only through global configuration mode. As an example, this chapter describes *interface configuration mode*, a commonly used configuration mode.

From configuration modes, you can enter configuration submodes. Configuration submodes are used for the configuration of specific features within the scope of a given configuration mode. As an example, this chapter describes the *subinterface configuration mode*, a submode of the interface configuration mode.

ROM monitor mode is a separate mode used when the router cannot boot properly. If your system (router, switch, or access server) does not find a valid system image to load when it is booting, the system will enter ROM monitor mode. ROM monitor (ROMMON) mode can also be accessed by interrupting the boot sequence during startup.

The following sections contain detailed information on these command modes:

Cisco IOS XE CLI Command Modes Overview, on page 2 follows these sections and summarizes the main Cisco IOS XE command modes.

Cisco IOS XE CLI Task List

To familiarize yourself with the features of the Cisco IOS XE CLI, perform any of the tasks described in the following sections:

Getting Context-Sensitive Help

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

To get help specific to a command mode, a command name, a keyword, or an argument, use any of the following commands:

Command	Purpose
(*prompt*)# **help**	Displays a brief description of the help system.
(*prompt*)# *abbreviated-command-entry***?**	Lists commands in the current mode that begin with a particular character string.
(*prompt*)# *abbreviated-command-entry* <**Tab**>	Completes a partial command name.
(*prompt*)# **?**	Lists all commands available in the command mode.
(*prompt*)# *command***?**	Lists the available syntax options (arguments and keywords) for the command.
(*prompt*)# *command keyword* **?**	Lists the next available syntax option for the command.

Note that the system prompt will vary depending on which configuration mode you are in.

When context-sensitive help is used, the space (or lack of a space) before the question mark (**?**) is significant. To obtain a list of commands that begin with a particular character sequence, type in those characters followed immediately by the question mark (**?**). Do not include a space. This form of help is called word help, because it completes a word for you. For more information, see the "Completing a Partial Command Name " section later in this chapter.

To list keywords or arguments, enter a question mark (**?**) in place of a keyword or argument. Include a space before the**?**. This form of help is called command syntax help, because it shows you which keywords or arguments are available based on the command, keywords, and arguments you already have entered.

You can abbreviate commands and keywords to the number of characters that allow a unique abbreviation. For example, you can abbreviate the **configureterminal**command to **configt**. Because the abbreviated form of the command is unique, the router will accept the abbreviated form and execute the command.

Entering the**help** command (available in any command mode) will provide the following description of the help system:

```
Router#
 help
Help may be requested at any point in a command by entering
a question mark '?'. If nothing matches, the help list will
be empty and you must back up until entering a '?' shows the
available options.
Two styles of help are provided:
1. Full help is available when you are ready to enter a
   command argument (e.g. 'show ?') and describes each possible
   argument.
2. Partial help is provided when an abbreviated argument is entered
   and you want to know what arguments match the input
   (e.g. 'show pr?'.)
```

As described in the **help** command output, you can use the question mark (**?**) to complete a partial command name (partial help), or to obtain a list of arguments or keywords that will complete the current command.

The following example illustrates how the context-sensitive help feature enables you to create an access list from configuration mode.

Enter the letters **co** at the system prompt followed by a question mark (**?**). Do not leave a space between the last letter and thequestion mark. The system provides the commands that begin with **co**.

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

Enter the **configure** command followed by a space and aquestion mark to list the keywords for the command and a brief explanation:

```
Router# configure ?
  memory    Configure from NV memory
  network   Configure from a TFTP network host
  overwrite-network  Overwrite NV memory from TFTP network host
  terminal  Configure from the terminal
  <cr>
```

The <cr> symbol (“cr” stands for carriage return) appears in the list to indicate that one of your options is to press the Return or Enter key to execute the command, without adding any keywords. In this example, the output indicates that your options for the configure command are **configurememory** (configure from NVRAM), **configurenetwork** (configure from a file on the network), **configureoverwrite-network** (configure from a file on the network and replace the file in NVRAM), or **configureterminal** (configure manually from the terminal connection). For most commands, the <cr> symbol is used to indicate that you can execute the command with the syntax you have already entered. However, the configure command is a special case, because the CLI will prompt you for the missing syntax:

```
Router# configure
Configuring from terminal, memory, or network [terminal]? terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
```

The default response for the ? prompt is indicated in the CLI output by a bracketed option at the end of the line. In the preceding example, pressing the Enter (or Return) key is equivalent to typing in the word “terminal.”

Enter the **configureterminal** command to enter global configuration mode:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

The CLI provides error isolation in the form of an error indicator, a caret symbol (^). The ^ symbol appears at the point in the command string where the user has entered incorrect or unrecognized command syntax. For example, the caret symbol in the following output shows the letter that was mistyped in the command:

```
Router# configure terminal
                   ^
% Invalid input detected at '^' marker.
Router#
```

Note that an error message (indicated by the % symbol) appears on the screen to alert you to the error marker.

Enter the **access-list** command followed by a space and a question mark to list the available options for the command:

```
Router(config)# access-list ?
 <1-99>             IP standard access list
 <100-199>          IP extended access list
 <1100-1199>        Extended 48-bit MAC address access list
 <1300-1999>        IP standard access list (expanded range)
 <200-299>          Protocol type-code access list
 <2000-2699>        IP extended access list (expanded range)
 <700-799>          48-bit MAC address access list
 dynamic-extended   Extend the dynamic ACL absolute timer
 rate-limit         Simple rate-limit specific access list
```

The two numbers within the angle brackets represent an inclusive range. Enter the access list number **99** and then enter another question mark to see the arguments that apply to the keyword and brief explanations:

```
Router(config)# access-list 99 ?
  deny    Specify packets to reject
  permit  Specify packets to forward
```

Enter the **deny** argument followed by a question mark (**?**)to list additional options:

```
Router(config)# access-list 99 deny ?
  A.B.C.D  Address to match
```

Generally, uppercase letters represent variables (arguments). Enter the IP address followed by a question mark (**?**) to list additional options:

```
Router(config)# access-list 99 deny 172.31.134.0 ?
  A.B.C.D  Mask of bits to ignore
  <cr>
```

In this output, A.B.C.D indicates that use of a wildcard mask is allowed. The wildcard mask is a method for matching IP addresses or ranges of IP addresses. For example, a wildcard mask of 0.0.0.255 matches any number in the range from 0 to 255 that appears in the fourth octet of an IP address.

Enter the wildcard mask followed by a question mark (**?**) to list further options:

```
Router(config)# access-list 99 deny 172.31.134.0 0.0.0.255 ?
<cr>
```

The <cr> symbol by itself indicates there are no more keywords or arguments. Press Enter (or Return) to execute the command.:

```
Router(config)# access-list 99 deny 172.31.134.0 0.0.0.255
```

The system adds an entry to access list 99 that denies access to all hosts on subnet 172.31.134.0, while ignoring bits for IP addresses that end in 0 to 255.

Using the no and default Forms of Commands

Almost every configuration command has a **no** form. In general, use the **no** form to disable a feature or function. Use the command without the **no**keyword to reenable a disabled feature or to enable a feature that

is disabled by default. For example, IP routing is enabled by default. To disable IP routing, use the **noiprouting** form of the **iprouting** command. To reenable it, use the plain **iprouting** form. The Cisco IOS software command reference publications describe the function of the **no** form of the command whenever a **no** form is available.

Many CLI commands also have a**default** form. By issuing the **default***command-name* command, you can configure the command to its default setting. The Cisco IOS software command reference documents generally 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.

Using Command History

The Cisco IOS CLI provides a history or record of commands that you have entered. This feature is particularly useful for recalling long or complex commands or entries, including access lists. To use the command history feature, perform any of the tasks described in the following sections:

Using CLI Editing Features and Shortcuts

A variety of shortcuts and editing features are enabled for the Cisco IOS CLI. The following subsections describe these features:

Moving the Cursor on the Command Line

The table below shows the key combinations or sequences you can use to move the cursor on the command line to make corrections or changes. Ctrl indicates the Control key, which must be pressed simultaneously with its associated letter key. Esc indicates the Escape key, which must be pressed first, followed by its associated letter key. Keys are not case sensitive. Many letters used for CLI navigation and editing were chosen to provide an easy way of remembering their functions. In the table below characters are bolded in the "Function Summary" column to indicate the relation between the letter used and the function.

Table 1: Key Combinations Used to Move the Cursor

Keystrokes	Function Summary	Function Details
Left Arrow or **Ctrl-B**	**B** ack character	Moves the cursor one character to the left. When you enter a command that extends beyond a single line, you can press the Left Arrow or Ctrl-B keys repeatedly to scroll back toward the system prompt and verify the beginning of the command entry, or you can press the Ctrl-A key combination.
Right Arrow or **Ctrl-F**	**F** orward character	Moves the cursor one character to the right.
Esc , **B**	**B** ack word	Moves the cursor back one word.

Keystrokes	Function Summary	Function Details
Esc , **F**	**F** orward word	Moves the cursor forward one word.
Ctrl -A	Beginning of line	Moves the cursor to the beginning of the line.
Ctrl -E	**E** nd of line	Moves the cursor to the end of the command line.

Completing a Partial Command Name

If you cannot remember a complete command name, or if you want to reduce the amount of typing you have to perform, enter the first few letters of the command, then press the Tab key. The command line parser will complete the command if the string entered is unique to the command mode. If your keyboard does not have a Tab key, press **Ctrl-I** instead.

The CLI will recognize a command once you have entered enough characters to make the command unique. For example, if you enter **conf** in privileged EXEC mode, the CLI will be able to associate your entry with the **configure** command, because only the **configure** command begins with **conf**.

In the following example the CLI recognizes the unique string for privileged EXEC mode of **conf**when the Tab key is pressed:

```
Router# conf
<Tab
>
Router# configure
```

When you use the command completion feature the CLI displays the full command name. The command is not executed until you use the Return or Enter key. This way you can modify the command if the full command was not what you intended by the abbreviation. If you enter a set of characters that could indicate more than one command, the system beeps to indicate that the text string is not unique.

If the CLI cannot complete the command, enter a question mark (**?**) to obtain a list of commands that begin with that set of characters. Do not leave a space between the last letter you enter and the question mark (**?**).

For example, entering **co?** will list all commands available in the current command mode:

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

Note that the characters you enter before the question mark appear on the screen to allow you to complete the command entry.

Recalling Deleted Entries

The CLI stores commands or keywords that you delete in a history buffer. Only character strings that begin or end with a space are stored in the buffer; individual characters that you delete (using Backspace or Ctrl-D) are not stored. The buffer stores the last ten items that have been deleted using Ctrl-K, Ctrl-U, or Ctrl-X. To recall these items and paste them in the command line, use the following key combinations:

Keystrokes	Purpose
Ctrl -Y	Recalls the most recent entry in the buffer (press keys simultaneously).
Esc , **Y**	Recalls the previous entry in the history buffer (press keys sequentially).

Note that the Esc, Y key sequence will not function unless you press the Ctrl-Y key combination first. If you press Esc, Ymore than ten times, you will cycle back to the most recent entry in the buffer.

Editing Command Lines that Wrap

The CLI provides a wrap-around feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command. To scroll back, press Ctrl-B or the Left Arrow key repeatedly until you scroll back to the beginning of the command entry, or press Ctrl-A to return directly to the beginning of the line.

In the following example, the **access-list** command entry extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign ($) indicates that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.

```
Router(config)# access-list 101 permit tcp 172.31.134.5 255.255.255.0 172.31.1
Router(config)# $ 101 permit tcp 172.31.134.5 255.255.255.0 172.31.135.0 255.25
Router(config)# $t tcp 172.31.134.5 255.255.255.0 172.31.135.0 255.255.255.0 eq
Router(config)#
$31.134.5 255.255.255.0 172.31.135.0 255.255.255.0 eq 45
```

When you have completed the entry, press **Ctrl-A** to check the complete syntax before pressing the Return key to execute the command. The dollar sign ($) appears at the end of the line to indicate that the line has been scrolled to the right:

```
Router(config)# access-list 101 permit tcp 172.31.134.5 255.255.255.0 172.31.1$
```

The Cisco IOS XE software assumes you have a terminal screen that is 80 columns wide. If you have a different screen-width, use the **terminal width** user EXEC command to set the width of your terminal.

Use line wrapping in conjunction with the command history feature to recall and modify previous complex command entries. See the Recalling Commands section in this chapter for information about recalling previous command entries.

Deleting Entries

Use any of the following keys or key combinations to delete command entries if you make a mistake or change your mind:

Keystrokes	Purpose
Delete or **Backspace**	Deletes the character to the left of the cursor.
Ctrl -D	Deletes the character at the cursor.

Keystrokes	Purpose
Ctrl -K	Deletes all characters from the cursor to the end of the command line.
Ctrl -U or **Ctrl-X**	Deletes all characters from the cursor to the beginning of the command line.
Ctrl -W	Deletes the word to the left of the cursor.
Esc , **D**	Deletes from the cursor to the end of the word.

Continuing Output at the --More-- Prompt

When you use the Cisco IOS XE CLI, output often extends beyond the visible screen length. For cases where output continues beyond the bottom of the screen, such as with the output of many **?**, **show**, or **more** commands, the output is paused and a --More-- prompt appears at the bottom of the screen. To resume output, press the Return key to scroll down one line, or press the Spacebar to display the next full screen of output.

Tip If output is pausing on your screen, but you do not see the --More-- prompt, try entering a lower value for the screen length using the **length** line configuration command or the **terminal length** privileged EXEC mode command. Command output will not be paused if the **length** value is set to zero.

For information about filtering output from the --More-- prompt, see the Searching and Filtering CLI Output module in this chapter.

Redisplaying the Current Command Line

If you are entering a command and the system suddenly sends a message to your screen, you can easily recall your current command line entry. To redisplay the current command line (refresh the screen), use either of the following key combinations:

Keystrokes	Purpose
Ctrl -L or **Ctrl-R**	Redisplays the current command line.

Transposing Mistyped Characters

If you have mistyped a command entry, you can transpose the mistyped characters. To transpose characters, use the following key combination:

Keystrokes	Purpose
Ctrl -T	Transposes the character to the left of the cursor with the character located to the right of the cursor.

Controlling Capitalization

You can capitalize or lowercase words or capitalize a set of letters with simple key sequences. Note, however, that Cisco IOS XE commands are generally case-insensitive, and are typically all in lowercase. To change the capitalization of commands, use any of the following key sequences:

Keystrokes	Purpose
Esc , **C**	Capitalizes the letter at the cursor.
Esc , **L**	Changes the word at the cursor to lowercase.
Esc , **U**	Capitalizes letters from the cursor to the end of the word.

Designating a Keystroke as a Command Entry

You can configure the system to recognize a particular keystroke (key combination or sequence) as command aliases. In other words, you can set a keystroke as a shortcut for executing a command. To enable the system to interpret a keystroke as a command, use the either of the following key combinations before entering the command sequence:

Keystrokes	Purpose
Ctrl -V or **Esc**,**Q**	Configures the system to accept the following keystroke as a user-configured command entry (rather than as an editing command).

Disabling and Reenabling Editing Features

The editing features described in the previous sections are automatically enabled on your system. However, there may be some unique situations that could warrant disabling these editing features. For example, you may have scripts that conflict with editing functionality. To globally disable editing features, use the following command in line configuration mode:

Command	Purpose
`Router(config-line)# no editing`	Disables CLI editing features for a particular line.

To disable the editing features for the current terminal session, use the following command in user EXEC mode:

Command	Purpose
`Router#` **no terminal editing**	Disables CLI editing features for the local line.

To reenable the editing features for the current terminal session, use the following command in user EXEC mode:

Command	Purpose
`Router#` **terminal editing**	Enables the CLI editing features for the current terminal session.

To reenable the editing features for a specific line, use the following command in line configuration mode:

Command	Purpose
`Router(config-line)#` **editing**	Enables the CLI editing features.

Searching and Filtering CLI Output

The Cisco IOS CLI provides ways of searching through large amounts of command output and filtering output to exclude information you do not need. These features are enabled for**show** and **more** commands, which generally display large amounts of data.

Note **Show** and **more** commands are always entered in user EXEC or privileged EXEC.

When output continues beyond what is displayed on your screen, the Cisco IOS CLI displays a --More-- prompt. Pressing Return displays the next line; pressing the Spacebar displays the next screen of output. The CLI String Search feature allows you to search or filter output from --More-- prompts.

Using the Cisco IOS XE CLI Examples

Determining Command Syntax and Using Command History Example

The CLI provides error isolation in the form of an error indicator, a caret symbol (^). The ^ symbol appears at the point in the command string where you have entered an incorrect command, keyword, or argument.

In the following example, suppose you want to set the clock. Use context-sensitive help to determine the correct command syntax for setting the clock.

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

The help output shows that the **set** keyword is required. Determine the syntax for entering the time:

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

Enter the current time:

```
Router# clock set 13:32:00
% Incomplete command.
```

The system indicates that you need to provide additional arguments to complete the command. Press Ctrl-P or the Up Arrow to automatically repeat the previous command entry. Then add a space and question mark (**?**) to reveal the additional arguments:

```
Router# clock set 13:32:00 ?
  <1-31>     Day of the month
  MONTH Month of the year
```

Now you can complete the command entry:

```
Router# clock set 13:32:00 February 01
                                   ^
% Invalid input detected at '^' marker.
```

The caret symbol (^) and help response indicate an error at 01. To list the correct syntax, enter the command up to the point where the error occurred and then enter a question mark (**?**):

```
Router# clock set 13:32:00 February ?
<1-31>     Day of the month
Router# clock set 13:32:00 February 23 ?
<1993-2035> Year
```

Enter the year using the correct syntax and press Enter or Return to execute the command:

```
Router# clock set 13:32:00 February 23 2001
```

Searching and Filtering CLI Output Examples

The following is partial sample output from the **morenvram:startup-config|begin** privileged EXEC mode command that begins unfiltered output with the first line that contains the regular expression ip. At the --More-- prompt, the user specifies a filter to exclude output lines that contain the regular expression ip.

```
Router# more nvram:startup-config | begin ip
address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
!
security passwords min-length 1
!
no aaa new-model
ip subnet-zero
no ip domain lookup
ip host sjc-tftp02 171.69.17.17
ip host sjc-tftp01 171.69.17.19
```

```
ip host dirt 171.69.1.129
!
!
multilink bundle-name authenticated
!
!
redundancy
 mode sso
!
!
bba-group pppoe global
!
!
interface GigabitEthernet0/0/0
 ip address 10.4.9.158 255.255.255.0
 media-type rj45
 speed 1000
 duplex full
 negotiation auto
 no cdp enable
!
interface GigabitEthernet0/0/1
 no ip address
 media-type rj45
 speed 1000
 duplex full
 negotiation auto
 no cdp enable
!
interface POS0/1/0
 no ip address
 shutdown
 no cdp enable
!
interface POS0/1/1
 no ip address
 shutdown
 no cdp enable
!
interface GigabitEthernet0
 vrf forwarding Mgmt-intf
 no ip address
 speed 1000
 duplex full
 negotiation auto
!
ip default-gateway 10.4.9.1
ip classless
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 GigabitEthernet0/0/0
ip route 171.69.0.0 255.255.0.0 10.4.9.1
!
no ip http server
no ip http secure-server
!
!
snmp mib bulkstat schema E0
snmp mib bulkstat schema IFMIB
snmp mib bulkstat transfer 23
snmp mib bulkstat transfer bulkstat1
!
!
control-plane
!
!
line con 0
 exec-timeout 30 0
 logging synchronous
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 privilege level 15
```

```
 password lab
 login
!
end
```

The following is partial sample output of the **morenvram:startup-config|include** privileged EXEC command. It only displays lines that contain the regular expression ip .

```
Router# more nvram:startup-config | include ip
ip subnet-zero
ip domain-name cisco.com
ip name-server 1192.168.48.48
ip name-server 172.16.2.132
```

The following is partial sample output from the **morenvram:startup-config|exclude**privileged EXEC command. It excludes lines that contain the regular expression service . At the --More-- prompt, the user specifies a filter with the regular expression Dialer1. Specifying this filter resumes the output with the first line that contains Dialer1.

```
Router# more nvram:startup-config | exclude service
!
version 12.2
!
hostname router
!
boot system flash
no logging buffered
!
ip subnet-zero
ip domain-name cisco.com
.
.
.
--More--
/Dialer1
filtering...
interface Dialer1
 no ip address
 no ip directed-broadcast
 dialer in-band
 no cdp enable
```

The following is partial sample output from the**showinterface** user EXEC or privileged EXEC command mode with an output search specified. The use of the keywords **beginFastEthernet** after the pipe begins unfiltered output with the first line that contains the regular expression Fast Ethernet . At the --More-- prompt, the user specifies a filter that displays only the lines that contain the regular expression **Serial** .

```
Router# show interface | begin FastEthernet
FastEthernet0/0 is up, line protocol is up
Hardware is Lance, address is 0060.837c.6399 (bia 0060.837c.6399)
  Description: ip address is 172.1.2.14 255.255.255.0
  Internet address is 172.1.2.14/24
.
.
.
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
--More--
+Serial
filtering...
Serial1 is up, line protocol is up
Serial2 is up, line protocol is up
Serial3 is up, line protocol is down
Serial4 is down, line protocol is down
Serial5 is up, line protocol is up
Serial6 is up, line protocol is up
Serial7 is up, line protocol is up
```

The following is partial sample output from the**showbuffers|exclude**command. It excludes lines that contain the regular expression **0 misses** . At the --More-- prompt, the user specifies a search that continues the filtered output beginning with the first line that contains Serial0.

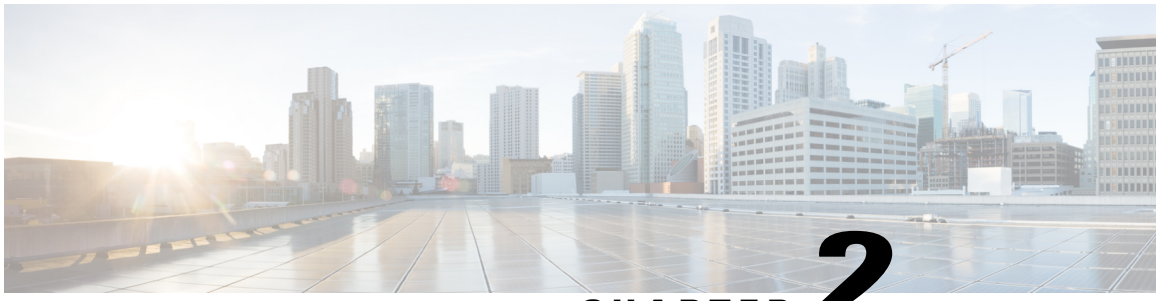
```
Router# show buffers | exclude 0 misses
Buffer elements:
     398 in free list (500 max allowed)
Public buffer pools:
Small buffers, 104 bytes (total 50, permanent 50):
     50 in free list (20 min, 150 max allowed)
     551 hits, 3 misses, 0 trims, 0 created
Big buffers, 1524 bytes (total 50, permanent 50):
     49 in free list (5 min, 150 max allowed)
Very Big buffers, 4520 bytes (total 10, permanent 10):
.
.
.
Huge buffers, 18024 bytes (total 0 permanent 0):
     0 in free list (0 min, 4 max allowed)
--More--
/Serial0
filtering...
Serial0 buffers, 1543 bytes (total 64, permanent 64):
     16 in free list (0 min, 64 max allowed)
     48 hits, 0 fallbacks
```

The following is partial sample output from the**showinterface|include**user EXEC or privileged EXEC command mode. The use of the **include(is)** keywords after the pipe (|) causes the command to display only lines that contain the regular expression (is). The parenthesis force the inclusion of the spaces before and after is. Use of the parenthesis ensures that only lines containing is with a space both before and after it will be included in the output (excluding from the search, for example, words like “disconnect”**) .**

```
router# show interface | include ( is )
ATM0 is administratively down, line protocol is down
  Hardware is ATMizer BX-50
Dialer0/1 is up (spoofing), line protocol is up (spoofing)
  Hardware is Unknown
  DTR is pulsed for 1 seconds on reset
FastEthernet0/0 is up, line protocol is up
  Hardware is Lance, address is 0060.837c.6399 (bia 0060.837c.6399)
  Internet address is 172.21.53.199/24
FastEthernet0/1 is up, line protocol is up
  Hardware is Lance, address is 0060.837c.639c (bia 0060.837c.639c)
  Internet address is 10.5.5.99/24
Serial0:0 is down, line protocol is down
  Hardware is DSX1
.
.
.
 --More--
```

At the --More-- prompt, the user specifies a search that continues the filtered output beginning with the first line that contains Serial0:13:

```
/Serial0:13
filtering...
Serial0:13 is down, line protocol is down
  Hardware is DSX1
  Internet address is 10.0.0.2/8
     0 output errors, 0 collisions, 2 interface resets
  Timeslot(s) Used:14, Transmitter delay is 0 flag
```



CHAPTER 2

Searching and Filtering CLI Output

The Cisco IOS CLI provides ways of searching through large amounts of command output and filtering output to exclude information you do not need. These features are enabled for**show** and **more** commands, which generally display large amounts of data.

Note **Show** and **more** commands are always entered in user EXEC or privileged EXEC.

When output continues beyond what is displayed on your screen, the Cisco IOS CLI displays a --More-- prompt. Pressing Return displays the next line; pressing the Spacebar displays the next screen of output. The CLI String Search feature allows you to search or filter output from --More-- prompts.

Finding Feature Information

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn . An account on Cisco.com is not required.

Understanding Regular Expressions

A regular expression is a pattern (a phrase, number, or more complex pattern) the CLI String Search feature matches against **show** or **more** command output. Regular expressions are case-sensitive and allow for complex matching requirements. Simple regular expressions include entries like Serial, misses, or 138. Complex regular expressions include entries like 00210... , (is), or [Oo]utput.

A regular expression can be a single-character pattern or a multiple-character pattern. That is, a regular expression can be a single character that matches the same single character in the command output or multiple characters that match the same multiple characters in the command output. The pattern in the command output is referred to as a string. This section describes creating both single-character patterns and multiple-character

patterns. It also discusses creating more complex regular expressions using multipliers, alternation, anchoring, and parentheses.

Single-Character Patterns

The simplest regular expression is a single character that matches the same single character in the command output. You can use any letter (A-Z, a-z) or digit (0-9) as a single-character pattern. You can also use other keyboard characters (such as ! or ~) as single-character patterns, but certain keyboard characters have special meaning when used in regular expressions. The table below lists the keyboard characters that have special meaning.

Table 2: Characters with Special Meaning

Character	Special Meaning
.	Matches any single character, including white space.
*	Matchers 0 or more sequences of the pattern.
+	Matches 1 or more sequences of the pattern.
?	Matches 0 or 1 occurrences of the pattern.
^	Matches the beginning of the string.
$	Matches the end of the string.
_ (underscore)	Matches a comma (,), left brace ({), right brace (}), left parenthesis (( ), right parenthesis ( )), the beginning of the string, the end of the string, or a space.

To use these special characters as single-character patterns, remove the special meaning by preceding each character with a backslash (\). The following examples are single-character patterns matching a dollar sign, an underscore, and a plus sign, respectively.

\$ _ \+

You can specify a range of single-character patterns to match against command output. For example, you can create a regular expression that matches a string containing one of the following letters: a, e, i, o, or u. Only one of these characters must exist in the string for pattern matching to succeed. To specify a range of single-character patterns, enclose the single-character patterns in square brackets (**[]**). For example, **[aeiou]**matches any one of the five vowels of the lowercase alphabet, while **[abcdABCD]** matches any one of the first four letters of the lower- or uppercase alphabet.

You can simplify ranges by entering only the endpoints of the range separated by a dash (-). Simplify the previous range as follows:

[a-dA-D]

To add a dash as a single-character pattern in your range, include another dash and precede it with a backslash:

[a-dA-D\-]

You can also include a right square bracket (]) as a single-character pattern in your range, as shown here:

[a-dA-D\-\]]

The previous example matches any one of the first four letters of the lower- or uppercase alphabet, a dash, or a right square bracket.

You can reverse the matching of the range by including a caret (^) at the start of the range. The following example matches any letter except the ones listed:

[^a-dqsv]

The following example matches anything except a right square bracket (]) or the letter d:

[^\]d]

Multiple-Character Patterns

When creating regular expressions, you can also specify a pattern containing multiple characters. You create multiple-character regular expressions by joining letters, digits, or keyboard characters that do not have special meaning. For example, a4% is a multiple-character regular expression. Insert a backslash before the keyboard characters that have special meaning when you want to indicate that the character should be interpreted literally.

With multiple-character patterns, order is important. The regular expression **a4%** matches the character a followed by a 4 followed by a % sign. If the string does not have a4%, in that order, pattern matching fails. The multiple-character regular expression **a.**uses the special meaning of the period character to match the letter a followed by any single character. With this example, the strings ab, a!, or a2 are all valid matches for the regular expression.

You can remove the special meaning of the period character by inserting a backslash before it. For example, when the expression **a\.** is used in the command syntax, only the string a. will be matched.

You can create a multiple-character regular expression containing all letters, all digits, all keyboard characters, or a combination of letters, digits, and other keyboard characters. For example, **telebit3107v32bis** is a valid regular expression.

Multipliers

You can create more complex regular expressions that instruct Cisco IOS software to match multiple occurrences of a specified regular expression. To do so, you use some special characters with your single-character and multiple-character patterns. The table below lists the special characters that specify “multiples” of a regular expression.

Table 3: Special Characters Used as Multipliers

Character	Description
*	Matches 0 or more single-character or multiple-character patterns.
+	Matches 1 or more single-character or multiple-character patterns.

Character	Description
?	Matches 0 or 1 occurrences of a single-character or multiple-character pattern.

The following example matches any number of occurrences of the letter a, including none:

a*

The following pattern requires that at least one letter a be in the string to be matched:

a+

The following pattern matches the string bb or bab:

ba?b

The following string matches any number of asterisks (*):

******

To use multipliers with multiple-character patterns, you enclose the pattern in parentheses. In the following example, the pattern matches any number of the multiple-character string ab:

(ab)*

As a more complex example, the following pattern matches one or more instances of alphanumeric pairs, but not none (that is, an empty string is not a match):

([A-Za-z][0-9])+

The order for matches using multipliers (*, +, or ?) is to put the longest construct first. Nested constructs are matched from outside to inside. Concatenated constructs are matched beginning at the left side of the construct. Thus, the regular expression matches A9b3, but not 9Ab3 because the letters are specified before the numbers.

Alternation

Alternation allows you to specify alternative patterns to match against a string. You separate the alternative patterns with a vertical bar (|). Exactly one of the alternatives can match the string. For example, the regular expression **codex|telebit** matches the string codex or the string telebit, but not both codex and telebit.

Anchoring

You can instruct Cisco IOS software to match a regular expression pattern against the beginning or the end of the string. That is, you can specify that the beginning or end of a string contain a specific pattern. You “anchor” these regular expressions to a portion of the string using the special characters shown in the table below.

Table 4: Special Characters Used for Anchoring

Character	Description
^	Matches the beginning of the string.
$	Matches the end of the string.

For example, the regular expression **^con**matches any string that starts with con, and **$sole** matches any string that ends with sole.

In addition to indicating the beginning of a string, the ^ symbol can be used to indicate the logical function "not" when used in a bracketed range. For example, the expression **[^abcd]**indicates a range that matches any single letter, as long as it is not the letters a, b, c, or d.

Contrast these anchoring characters with the special character underscore (_). Underscore matches the beginning of a string (^), the end of a string ($), parentheses (( )), space ( ), braces ({}), comma (,), or underscore (_). With the underscore character, you can specify that a pattern exist anywhere in the string. For example, **_1300_**matches any string that has 1300 somewhere in the string. The string 1300 can be preceded by or end with a space, brace, comma, or underscore. So, although {1300_matches the regular expression **_1300_**, 21300 and 13000 do not.

Using the underscore character, you can replace long regular expression lists. For example, instead of specifying **^1300()()1300${1300,,1300,{1300},1300,(1300**you can specify simply **_1300_**.

Parentheses for Recall

As shown in the "Multipliers" section, you use parentheses with multiple-character regular expressions to multiply the occurrence of a pattern. You can also use parentheses around a single- or multiple-character pattern to instruct the Cisco IOS software to remember a pattern for use elsewhere in the regular expression.

To create a regular expression that recalls a previous pattern, you use parentheses to indicate memory of a specific pattern and a backslash (\) followed by a number to reuse the remembered pattern. The number specifies the occurrence of a parentheses in the regular expression pattern. If you have more than one remembered pattern in your regular expression, then \1 indicates the first remembered pattern, and \2 indicates the second remembered pattern, and so on.

The following regular expression uses parentheses for recall:

a(.)bc(.)\1\2

This regular expression matches an a followed by any character (call it character no. 1), followed by bc followed by any character (character number 2), followed by character no. 1 again, followed by character number. 2 again. So, the regular expression can match aZbcTZT. The software remembers that character number 1 is Z and character number 2 is T and then uses Z and T again later in the regular expression.

Searching and Filtering show Commands

To searc h **show** command output, use the following command in privileged EXEC mode:

Command	Purpose
`Router#` **show** *`any-command`* | **begin** *`regular-expression`*	Begins unfiltered output of the **show** command with the first line that contains the regular expression.

Note Cisco IOS documentation generally uses the vertical bar to indicate a choice of syntax. However, to search the output of **show** and **more** commands, you will need to enter the pipe character (the vertical bar). In this section the pipe appears in bold (|) to indicate that you should enter this character.

To filter **show** command output, use one of the following commands in privileged EXEC mode:

Command	Purpose
`Router#` **show** *any-command* **|** **exclude** *regular-expression*	Displays output lines that do not contain the regular expression.
`Router#` **show** *any-command* **|** **include** *regular-expression*	Displays output lines that contain the regular expression.

On most systems you can enter the Ctrl-Z key combination at any time to interrupt the output and return to privileged EXEC mode. For example, you can enter the **showrunning-config|beginhostname** command to start the display of the running configuration file at the line containing the hostname setting, then use Ctrl-Z when you get to the end of the information you are interested in.

Note Characters followed by an exclamation mark (!) or a semicolon (;) are considered as a comment and hence they are ignored in a command.

Searching and Filtering more Commands

You can search **more** commands the same way you search **show** commands (**more** commands perform the same function as **show** commands). To search **more** command output, use the following command in user EXEC mode:

Command	Purpose
`Router#` **more** *any-command* **|** **begin** *regular-expression*	Begins unfiltered output of a **more** command with the first line that contains the regular expression.

You can filter **more** commands the same way you filter **show** commands. To filter **more** command output, use one of the following commands in user EXEC mode:

Command	Purpose
`Router#` **more** *any-command* **|** **exclude** *regular-expression*	Displays output lines that do not contain the regular expression.
`Router#` **more** *any-command* **|** **include** *regular-expression*	Displays output lines that contain the regular expression.

Searching and Filtering from the --More--Prompt

You can search output from --More-- prompts. To search **show** or **more** command output from a --More-- prompt, use the following command in user EXEC mode:

Command	Purpose
-More- / *regular-expression*	Begins unfiltered output with the first line that contains the regular expression.

You can filter output from --More-- prompts. However, you can specify only one filter for each command. The filter remains until the **show** or **more** command output finishes or until you interrupt the output (usingCtrl-Z or Ctrl-6). Therefore, you cannot add a second filter at a --More-- prompt if you already specified a filter at the original command or at a previous --More--prompt.

Note Searching and filtering are different functions. You can search command output using the **begin** keyword and specify a filter at the --More-- prompt for the same command.

To filter **show** or **more** command output at a --More-- prompt, use one of the following commands in user EXEC mode:

Command	Purpose
-More- - *regular-expression*	Displays output lines that do not contain the regular expression.
-More- + *regular-expression*	Displays output lines that contain the regular expression.

Searching and Filtering CLI Output Examples

The following is partial sample output from the **morenvram:startup-config|begin** privileged EXEC mode command that begins unfiltered output with the first line that contains the regular expression ip. At the --More-- prompt, the user specifies a filter to exclude output lines that contain the regular expression ip.

```
Router# more nvram:startup-config | begin ip
ip subnet-zero
ip domain-name cisco.com
ip name-server 192.168.48.48
ip name-server 172.16.2.132
!
isdn switch-type primary-5ess
.
.
.
interface Ethernet1
 ip address 10.5.5.99 10.255.255.0
 --More--
-ip
filtering...
 media-type 10BaseT
!
interface Serial0:23
 encapsulation frame-relay
 no keepalive
 dialer string 4001
 dialer-group 1
 isdn switch-type primary-5ess
 no fair-queue
```

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

```
Router# more nvram:startup-config | include ip
ip subnet-zero
ip domain-name cisco.com
ip name-server 1192.168.48.48
ip name-server 172.16.2.132
```

The following is partial sample output from the **morenvram:startup-config|exclude**command. It excludes lines that contain the regular expression service. At the --More-- prompt, the user specifies a filter with the regular expression Dialer1. Specifying this filter resumes the output with the first line that contains Dialer1.

```
Router# more nvram:startup-config | exclude service
!
version 12.2
!
hostname router
!
boot system flash
no logging buffered
!
ip subnet-zero
ip domain-name cisco.com
.
.
.
--More--
/Dialer1
filtering...
interface Dialer1
 no ip address
 no ip directed-broadcast
 dialer in-band
 no cdp enable
```

The following is partial sample output from the**showinterface** command with an output search specified. The use of the keywords **beginEthernet** after the pipe begins unfiltered output with the first line that contains the regular expression Ethernet. At the --More-- prompt, the user specifies a filter that displays only the lines that contain the regular expression Serial.

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

The following is partial sample output from the**showbuffers|exclude**command. It excludes lines that contain the regular expression ip. At the --More-- prompt, the user specifies a search that continues the filtered output beginning with the first line that contains Serial0.

```
Router# show buffers | exclude 0 misses
Buffer elements:
     398 in free list (500 max allowed)
Public buffer pools:
Small buffers, 104 bytes (total 50, permanent 50):
     50 in free list (20 min, 150 max allowed)
     551 hits, 3 misses, 0 trims, 0 created
Big buffers, 1524 bytes (total 50, permanent 50):
     49 in free list (5 min, 150 max allowed)
Very Big buffers, 4520 bytes (total 10, permanent 10):
.
.
.
Huge buffers, 18024 bytes (total 0 permanent 0):
     0 in free list (0 min, 4 max allowed)
--More--
/Serial0
filtering...
Serial0 buffers, 1543 bytes (total 64, permanent 64):
     16 in free list (0 min, 64 max allowed)
     48 hits, 0 fallbacks
```

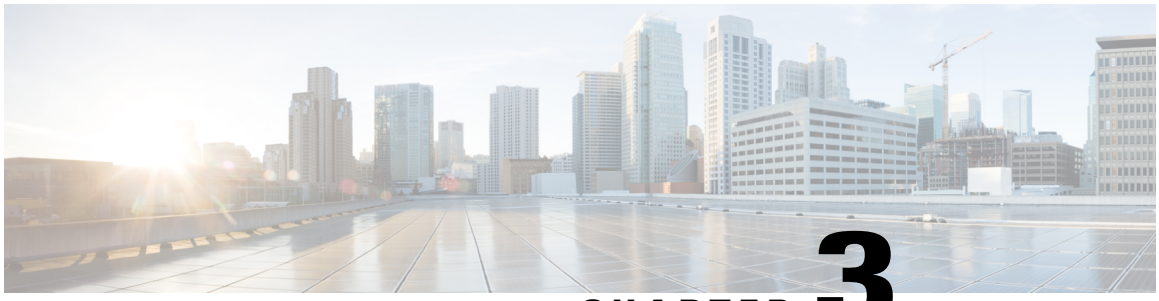
The following is partial sample output from the**showinterface|include**command. The use of the **include(is)** keywords after the pipe (|) causes the command to display only lines that contain the regular expression (is). The parenthesis force the inclusion of the spaces before and after is. Use of the parenthesis ensures that only lines containing is with a space both before and after it will be included in the output (excluding from the search, for example, words like “disconnect”).

```
router# show interface | include ( is )
ATM0 is administratively down, line protocol is down
  Hardware is ATMizer BX-50
Dialer1 is up (spoofing), line protocol is up (spoofing)
  Hardware is Unknown
  DTR is pulsed for 1 seconds on reset
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0060.837c.6399 (bia 0060.837c.6399)
  Internet address is 172.21.53.199/24
Ethernet1 is up, line protocol is up
```

```
  Hardware is Lance, address is 0060.837c.639c (bia 0060.837c.639c)
  Internet address is 10.5.5.99/24
Serial0:0 is down, line protocol is down
  Hardware is DSX1
.
.
.
 --More--
```

At the --More-- prompt, the user specifies a search that continues the filtered output beginning with the first line that contains Serial0:13:

```
/Serial0:13
filtering...
Serial0:13 is down, line protocol is down
  Hardware is DSX1
  Internet address is 10.0.0.2/8
     0 output errors, 0 collisions, 2 interface resets
  Timeslot(s) Used:14, Transmitter delay is 0 flag
```



CHAPTER 3

EXEC Commands in Configuration Mode

Beginning in Cisco IOS Release 12.1(11b)E, EXEC-level Cisco IOS commands (such as **show**, **clear**, and **debug** commands) can be entered within any configuration mode (such as global configuration mode) by issuing the **do**command followed by the desired EXEC command. This feature provides the convenience of entering EXEC-level commands without needing to exit the current configuration mode.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for EXEC Commands in Configuration Mode

You must have your network up and running with Cisco IOS Release 12.1(11b)E or a later release installed.

How to Enter EXEC Commands in Configuration Mode

Using the do Command in Configuration Mode

To execute an EXEC-level command in any configuration mode (including configuration submodes), complete the tasks in this section:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **do** *command*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**do** *command* **Example:** `Router(config)# configuration command`	Allows you to execute any EXEC mode command from within any configuration mode. *command* --The EXEC command to be executed.

Using the do Command in Interface Configuration Mode

To execute an EXEC-level command for a specific interface on a router, complete the task in this section:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot /port*
4. **do** *command*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type slot /port* **Example:** `Router(config)# interface serial 3/0`	The syntax for this command varies according to your platform and Cisco IOS release. For complete information, refer to the "Additional References" section. • The slot/port argument identifies the slot and port on the router where you are entering **do** commands.
Step 4	**do** *command* **Example:** `Router(config-if)# do show interfaces serial 3/0`	Allows you to execute any EXEC mode command from within any configuration mode on a specific interface. *command* --The EXEC command to be executed.

Configuration Examples for EXEC Commands in Configuration Mode

Example do show interface Command

The following example shows how to execute the EXEC-level **showinterface** command from within global configuration mode:

```
Router(config)# do show interfaces serial 3/0
Serial3/0 is up, line protocol is up
  Hardware is M8T-RS232
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input never, output 1d17h, output hang never
  Last clearing of "show interface" counters never
.
.
```

Example do clear vpdn tunnel Command

The following example shows how to execute the EXEC-level **clearvpdntunnel** command from within VPDN configuration mode:

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

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Cisco IOS configuration commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified	--

MIBs

MIB	MIBs Link
• No new or modified MIBs are supported, and support for existing MIBs has not been modified.	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

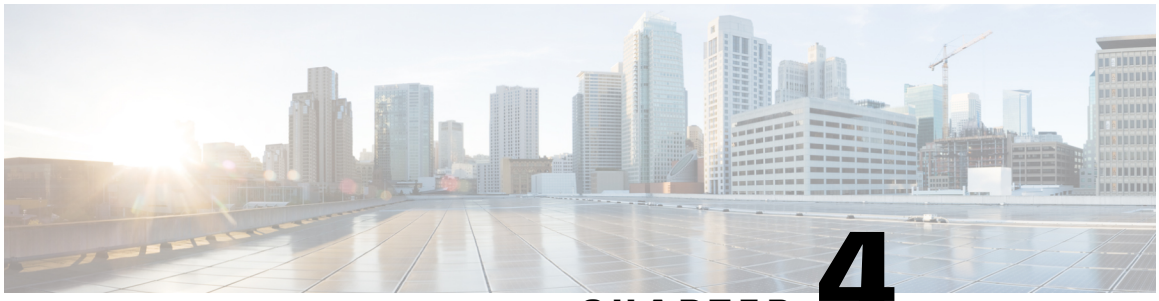
RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Restrictions for EXEC Commands in Configuration Mode

You cannot use the **do** command to execute the **configureterminal** EXEC command because issuing the **configureterminal**command changes the mode to configuration mode.



CHAPTER 4

show Command Output Redirection

The show Command Output Redirection feature provides the capability to redirect output from Cisco IOS command-line interface (CLI) **show** commands and **more** commands to a file.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About show Command Output Redirection

This feature enhances the **show** commands in the Cisco IOS CLI to allow large amounts of data output to be written directly to a file for later reference. This file can be saved on local or remote storage devices such as Flash, a SAN Disk, or an external memory device.

For each **show** command issued, a new file can be created, or the output can be appended to an existing file. Command output can optionally be displayed on-screen while being redirected to a file by using the **tee** keyword. Redirection is available using a pipe (|) character after any **show** command, combined with the followingkeywords:

Output redirection keywords:

Keyword	Usage
append	Append redirected output to URL (URLs supporting append operation only)
begin	Begin with the line that matches
count	Count number of lines which match regexp
exclude	Exclude lines that match
format	Format the output using the specified spec file
include	Include lines that match
redirect	Redirect output to URL
tee	Copy output to URL

These extenstions can also be added to **more** commands.

How to Use the show Command Enhancement

No configuration tasks are associated with this enhancement. For usage guidelines, see the command reference documents listed in the “Related Documents” section.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Cisco IOS configuration commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standard	Title
No new or modified standards are supported, and support for existing standards has not been modified	--

MIBs

MIB	MIBs Link
• No new or modified MIBs are supported, and support for existing MIBs has not been modified.	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported, and support for existing RFCs has not been modified.	--

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

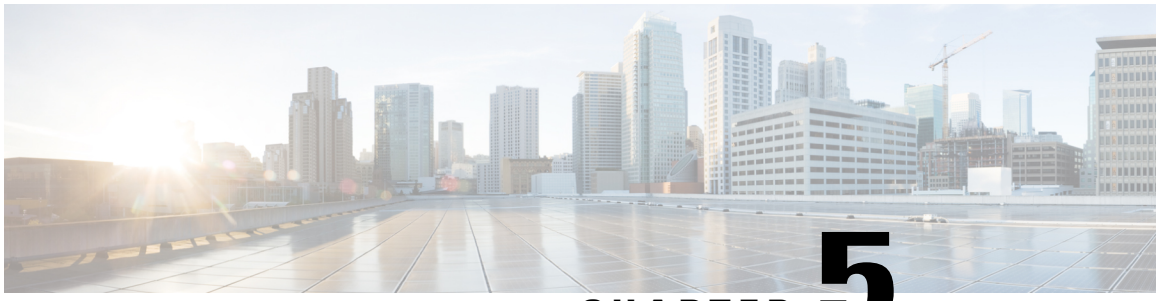
Feature Information for show Command Output Redirection

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 5: Feature Information for the show Command Ouput Redirection Feature

Feature Name	Releases	Feature Information
show Command Output Redirection	12.0(21)S 12.2(13)T	• The show Command Output Redirection feature provides the capability to redirect output from Cisco IOS command-line interface (CLI) **show** commands and **more** commands to a file. The following commands were introduced or modified: **show**, and**more**.



CHAPTER 5

Overview Basic Configuration of a Cisco Networking Device

Cisco IOS software provides two features, AutoInstall and Setup mode, to simplify configuring a Cisco IOS-based networking device. AutoInstall enables automatic loading of device configuration files from a remote location and can be used to configure several devices concurrently. Setup is an interactive Cisco IOS software command-line interface (CLI) mode that guides you through a basic (also called a startup) configuration but limits you to configuring a single device at a time. AutoInstall is an automatic process for the device that is being configured; Setup is a manual process for the device that is being configured.

This module provides an introduction to each feature and directs you to modules that describe the features in detail and explain how to use them.

The terms initial configuration and startup configuration are used interchangeably.

Prerequisites for Basic Configuration of a Cisco Networking Device

Prerequisites for Cisco IOS AutoInstall

- Using AutoInstall to Remotely Configure Cisco Networking Devices module is written specifically for networking devices running Cisco IOS Release 12.4(1) or newer. However most of the information in this document can be used to configure networking devices that support AutoInstall and are not running Cisco IOS release 12.4(1) or newer. The two key differences that you must allow for are:

- Some Cisco networking devices use BOOTP instead of DHCP to request IP address addresses over LAN interfaces. Enabling BOOTP support on your DHCP server will resolve this issue.
- Some Cisco networking devices use a DHCP client identifier format that is different from the format used by networking devices running Cisco IOS release 12.4(1) or newer. This document only explains the DHCP client identifier format used by networking devices running Cisco IOS release 12.4(1) or newer. Use the process described in the "Determining the Value for the DHCP Client Identifier Automatically" section in Using AutoInstall to Remotely Configure Cisco Networking Devices module to determine the DHCP client identifier format that your Cisco networking device is using.

- No configuration file resides in NVRAM on the networking device that is being configured with AutoInstall.
- The configuration files that you want to load on to the networking device using AutoInstall reside on a TFTP server that is connected to the network. In most cases there is more than one file; for example, a network file with the IP-to-hostname mappings and a device-specific configuration file.
- You have someone at the remote site to connect the networking device that is being configured with AutoInstall to the network and power it on.
- The network has the IP connectivity necessary to permit the networking device to load configuration files from the TFTP server during the AutoInstall process.
- A DHCP server is available on the network to provide IP addresses to networking devices that are using AutoInstall over a LAN connection.

Prerequisites for Cisco IOS Setup Mode

- A terminal is connected to the console port of the device being configured.
- You know the interfaces you want to configure.
- You know the routing protocols you want to enable.

For information about routing protocols, see the *Cisco IOS IP Routing Protocols Configuration Guide* .

- You know whether the device you are configuring will perform bridging.
- You know whether the device you are configuring has protocol translation installed.
- You have network addresses for the protocols being configured.

For information about network addresses, see the *Cisco IOS IP Addressing Services Configuration Guide*.

- You have a password strategy for your network environment.

For information about passwords and device security, see "Configuring Security with Passwords, Privilege Levels, and Login User names for CLI Sessions on Networking Devices" in the *Cisco IOS Security Configuration Guide* .

- You have or have access to documentation for the product you want to configure.

Restrictions for Basic Configuration of a Cisco Networking Device

Restrictions for Cisco IOS AutoInstall

- (Serial interfaces only) AutoInstall over a serial interface using either HDLC or Frame Relay can be performed only over the first serial port on a new device (serial interface 0 or serial interface x/0).
- (LAN interfaces only) Only LAN Token Ring interfaces that set ring speed with physical jumpers support AutoInstall.

Restrictions for Cisco IOS Setup Mode

- Setup mode is hardware dependent. You must follow instructions for the specific product you want to configure, as described in documentation for that product.
- Some configuration parameters apply only when a networking device has the protocol translation option. If a device does not have protocol translation, Setup does not prompt for these parameters.

Information About Basic Configuration of a Cisco Networking Device

Before you configure a networking device with a basic configuration, you should understand the following concepts and decide whether AutoInstall or Setup mode is the best method, based on your requirements.

Comparison of Cisco IOS AutoInstall and Cisco IOS Setup Mode

Cisco IOS AutoInstall enables automatic loading of device configuration files from a remote location and can be used to configure several devices concurrently. Setup is an interactive Cisco IOS software CLI mode that guides you through a basic (also called a startup) configuration but limits you to configuring a single device at a time. AutoInstall is an automatic process; Setup is a manual process.

Cisco IOS AutoInstall

AutoInstall is the Cisco IOS software feature that enables the configuration of a remote networking device from a central location. The configuration files must be stored on a TFTP server that is accessible by the devices that you are using AutoInstall to setup.

AutoInstall is supported over Ethernet, Token Ring, and FDDI interfaces for LANs, serial interfaces using High-Level Data Link Control (HDLC) encapsulation, serial interfaces using Frame Relay encapsulation for WANs, and WIC-1-DSU-T1v2 cards (No other T1E1 card supports Autoinstall.).

AutoInstall is designed to facilitate central management of installations at remote sites. The AutoInstall process begins when a Cisco IOS software-based device is turned on and a valid configuration file is not found in NVRAM. AutoInstall may not start if the networking device has Cisco Router and Security Device Manager (SDM) or Cisco Network Assistant already installed. In this case, to enable AutoInstall you need to disable SDM.

Using AutoInstall to Remotely Configure Cisco Networking Devices module describes how AutoInstall functions, how to disable SDM, and how to configure devices to use AutoInstall.

Cisco IOS Setup Mode

Cisco IOS Setup mode enables you to build an initial configuration file using the Cisco IOS CLI or System Configuration Dialog. The dialog guides you through initial configuration and is useful when you are unfamiliar with Cisco products or the CLI and when configuration changes do not require the level of detail the CLI provides.

Setup starts automatically when a device has no configuration file in NVRAM and is not preconfigured from the factory to use Cisco SDM. When setup completes, it presents the System Configuration Dialog. This dialog guides you through an initial configuration with prompts for basic information about your device and network and then creates an initial configuration file. After the file is created, you can use the CLI to perform additional configuration.

Using Setup Mode to Configure a Cisco Networking Device describes how to use Setup to build a basic configuration and to make configuration changes.

Where to Go Next

Proceed to either Using AutoInstall to Remotely Configure Cisco Networking Devices module or Using Setup Mode to Configure a Cisco Networking Device.

Additional References

This section provides references related to the basic configuration of a Cisco networking device.

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Configuration fundamentals commands	*Cisco IOS Configuration Fundamentals Command Reference*
Configuring a networking device for the first time using the Cisco IOS software feature AutoInstall.	Using AutoInstall to Remotely Configure Cisco Networking Devices module in *Cisco IOS Configuration Fundamentals Configuration Guide*
Configuring a networking device using Cisco IOS Setup mode	Using Setup Mode to Configure a Cisco Networking Device module in *Cisco IOS Configuration Fundamentals Configuration Guide*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

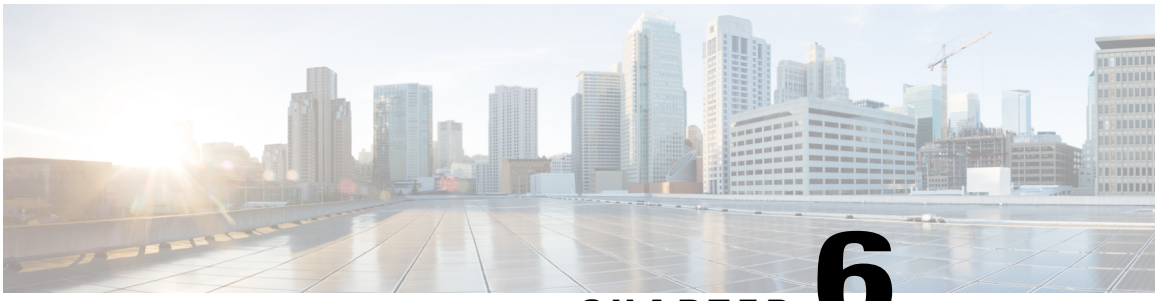
Feature Information for Overview Basic Configuration of a Cisco Networking Device

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 6: Feature Information for Overview: Basic Configuration of a Cisco Networking Device

Feature Name	Releases	Feature Information
Overview: Basic Configuration of a Cisco Networking Device	12.4(3)	Cisco IOS software provides two features, AutoInstall and Setup mode, to simplify configuring a Cisco IOS-based networking device. AutoInstall enables automatic loading of device configuration files from a remote location and can be used to configure several devices concurrently. Setup is an interactive Cisco IOS software command-line interface (CLI) mode that guides you through a basic (also called a startup) configuration but limits you to configuring a single device at a time. AutoInstall is an automatic process for the device that is being configured; Setup is a manual process for the device that is being configured.



CHAPTER 6

Using Setup Mode to Configure a Cisco Networking Device

Setup mode provides an interactive menu to help you to create an initial configuration file for a new networking device, or a device that you have erased the startup-config file from NVRAM. The nteractive menu guides you through initial configuration and is useful when you are unfamiliar with Cisco products or the command line interface (CLI) and when configuration changes do not require the level of detail the CLI provides. Setup mode can also be used to modify an existing configuration.

This section describes how to use the System Configuration Dialog to prepare a Cisco networking device for full configuration and how you can make configuration changes after an initial configuration is complete. To improve readability, filenames are enclosed in quotation marks. Also, the terms device and networking device mean a router, switch, or other device running Cisco IOS software. The terms initial configuration and startup configuration are used interchangeably.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Using Cisco IOS Setup Mode to Configure a Cisco Networking Device

- You have read the “Basic Configuration of a Cisco Networking Device Overview” module.
- An ASCII terminal is connected to the console port of the device being configured.
- You know the interfaces you want to configure.
- You know the routing protocols you want to enable.

For information about routing protocols, see the *Cisco IOS IP Routing Protocols Configuration Guide* , Release 12.4.

- You know whether the device you are configuring will perform bridging.
- You know whether the device you are configuring has protocol translation installed.
- You have network addresses for the protocols being configured.

For information about network addresses, see the *Cisco IOS IP Addressing Services Configuration Guide*, Release 12.4.

- You have a password strategy for your network environment.

For information about passwords and device security, see “Configuring Security with Passwords, Privilege Levels, and Login User names for CLI Sessions on Networking Devices” module in the *Cisco IOS Security Configuration Guide*, Release 12.4.

- You have or have access to documentation for the product you want to configure.

Restrictions for Using Cisco IOS Setup Mode to Configure a Cisco Networking Device

- Setup mode is hardware dependent. You must follow instructions for the specific product you want to configure, as described in documentation for that product.
- Some configuration parameters apply only when a networking device has the protocol translation option. If a device does not have protocol translation, Setup does not prompt for these parameters.

Information About Using Cisco IOS Setup Mode to Configure a Cisco Networking Device

Cisco IOS Setup Mode

Cisco IOS Setup mode enables you to build an initial configuration file using the Cisco IOS CLI or System Configuration Dialog. The dialog guides you through initial configuration and is useful when you are unfamiliar with Cisco products or the CLI and when configuration changes do not require the level of detail the CLI provides.

Setup starts automatically when a device has no configuration file in NVRAM and is not preconfigured from the factory to use Cisco Router and Security Device Manager (SDM). When setup completes, it presents the System Configuration Dialog. This dialog guides you through an initial configuration with prompts for basic information about your device and network and then creates an initial configuration file. After the file is created, you can use the CLI to perform additional configuration.

Cisco Router and Security Device Manager

Cisco SDM is a web-based device management tool for configuring Cisco IOS network connections and security features on networking devices. SDM provides a default configuration and various wizards to guide you step by step through configuring a Cisco networking device, additional LAN or WAN connections, and VPN connections; creating firewalls; and performing security audits.

In addition to building an initial configuration, SDM provides an Advanced Mode through which you can configure advanced features such as Firewall Policy and Network Address Translation (NAT).

Some Cisco products ship from the factory with SDM installed. If SDM is preinstalled on your device and you want to use Setup to configure an initial configuration, you first must disable the SDM default configuration.

System Configuration Dialog

The *System Configuration Dialog* is an interactive CLI mode that prompts you for information needed to build an initial configuration for a Cisco networking device. Like the CLI, the System Configuration Dialog provides help text at each prompt. To access this help text, you enter a question mark (**?**) at the prompt.

The prompts in the System Configuration Dialog vary depending on hardware, installed interface modules, and software image. To use the dialog for an initial configuration, you need to refer to product-specific documentation.

The values shown in square brackets next to prompts reflect the current settings. These may be default settings from the factory or the latest settings configured on the device. To accept these settings, you press **Enter** on the keyboard.

You can exit (**Ctrl-C**) the System Configuration Dialog and return to privileged EXEC mode without making changes and without going through the entire dialog. If you exit the dialog but want to continue with setup, you can issue the **setup** command in privileged EXEC mode.

When you complete all the steps in the dialog, the device displays the modified configuration file and asks if you want to use that file. You must answer yes or no; there is no default for this prompt. If you answer yes,

the file is saved to NVRAM as the startup configuration. If you answer no, the file is not saved and you must start at the beginning of the dialog if you want to build another initial configuration.

In addition to being a quick and easy way to perform an initial configuration, the System Configuration Dialog also is useful for performing basic configuration changes after an initial configuration has been performed.

Benefits of Using Cisco IOS Setup Mode

The System Configuration Dialog in Cisco IOS Setup mode can be a valuable tool for users who are unfamiliar with Cisco products or the CLI. The dialog guides users through the configuration process with prompts for basic information to get the device operational. When general configuration changes are needed, the dialog also is an alternative method to the detail-level CLI.

How to Use Cisco IOS Setup Mode to Configure a Cisco Networking Device and Make Configuration Changes

This section describes how to use the System Configuration Dialog to build an initial configuration file and to make configuration changes after a startup configuration has been loaded.

Disabling the SDM Default Configuration File

Perform this task if SDM was preinstalled on your device and you want to use Setup to build an initial configuration file. SDM remains on the device.

Perform this task if SDM was pre installed on your device and you want to use AutoInstall to configure the device instead. SDM remains on the device.

SUMMARY STEPS

1. Connect the console cable, shipped with your device, from the console port on the device to a serial port on your PC. Refer to the hardware installation guide for the device for instructions.
2. Connect the power supply to the device, plug the power supply into a power outlet, and turn on the device. Refer to the quick start guide for the device for instructions.
3. Use Hyperterminal or a similar terminal emulation program on your PC, with the following terminal emulation settings, to connect to the device:
4. **enable**
5. **erase startup-config**
6. **reload**

DETAILED STEPS

Step 1 Connect the console cable, shipped with your device, from the console port on the device to a serial port on your PC. Refer to the hardware installation guide for the device for instructions.

Step 2 Connect the power supply to the device, plug the power supply into a power outlet, and turn on the device. Refer to the quick start guide for the device for instructions.

Step 3 Use Hyperterminal or a similar terminal emulation program on your PC, with the following terminal emulation settings, to connect to the device:

- 9600 baud
- 8 data bits, no parity, 1 stop bit
- No flow control

Step 4 **enable**
Enter privileged EXEC mode.

enable

Example:

```
Router> enable
Router#
```

Step 5 **erase startup-config**
Erases the existing configuration in NVRAM.

Example:

```
Router# erase startup-config
```

Step 6 **reload**
Initiates the reload process. The router will initiate the AutoInstall process after it finishes the reload process.

Example:

```
Router# reload
```

Using the System Configuration Dialog to Create an Initial Configuration File

Perform this task to create an initial configuration for a Cisco networking device.

Before You Begin

If SDM is installed, you must disable its default configuration file before using Setup.

Note The System Configuration Dialog does not allow you to randomly select or enter parameters for configuration. You must move through the dialog step by step until the screen shows the information you want to change.

SUMMARY STEPS

1. **Power on the device.**
2. **Enter yes at the prompt to enter the initial configuration dialogue.**
3. **If you are prompted to continue with the configuration dialogue, enter yes** at**thepromptocontinuethedialog(thisstepmightnotappear)**.
4. The basic management screen is displayed:
5. Enter a hostname for the device. This example uses Router.
6. Enter an enable secret password. This password is encrypted and cannot be seen when viewing the configuration.
7. Enter an enable password that is different from the enable secret password. An enable password is not encrypted and can be seen when viewing the configuration:
8. Enter a virtual terminal password. This password allows access to the device through only the console port.
9. Respond to the following prompts as appropriate for your network. In this example, the current setting [no] is accepted by pressing **Enter**.
10. Select an interface to connect the router to the management network:
11. Respond to the prompts as appropriate for your network. In this example, IP is configured: an IP address is entered and the current subnet mask is accepted. The screen displays the command script created.
12. Enter **2orpressEnter**to save the configuration file to NVRAM and exit.

DETAILED STEPS

Step 1 **Power on the device.**

Step 2 **Enter yes at the prompt to enter the initial configuration dialogue.**
If the following messages appear at the end of the startup sequence, the System Configuration Dialog was invoked automatically:

Example:

```
--- System Configuration Dialog ---
Would you like to enter the initial configuration dialog? [yes/no]: yes
```

The screen displays the following:

Example:

```
--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]:
```

Step 3 **If you are prompted to continue with the configuration dialogue, enter yes** at**the prompt to continue the dialog (this step might not appear)**.

Example:

```
Continue with configuration dialog? [yes/no]: yes
```

Step 4 The basic management screen is displayed:

Example:

```
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]:
```

Enter **yes** to enter basic management setup:

Example:

```
Would you like to enter basic management setup? [yes/no]: yes
The screen displays the following:
Configuring global parameters:
Enter host name [R1]:
```

Step 5 Enter a hostname for the device. This example uses Router.

Example:

```
Configuring global parameters:
Enter host name [R1]: Router
The screen displays the following:
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:
```

Step 6 Enter an enable secret password. This password is encrypted and cannot be seen when viewing the configuration.

Example:

```
Enter enable secret: 1g2j3mm
```

The screen displays the following:

Example:

```
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:
```

Step 7 Enter an enable password that is different from the enable secret password. An enable password is not encrypted and can be seen when viewing the configuration:

Example:

```
  Enter enable password: cts54tnl
```

The screen displays the following:

Example:

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

Step 8 Enter a virtual terminal password. This password allows access to the device through only the console port.

Example:

```
  Enter virtual terminal password: tls6gato
```

The screen displays the following:

Example:

```
Configure SNMP Network Management? [no]:
```

Step 9 Respond to the following prompts as appropriate for your network. In this example, the current setting [no] is accepted by pressing **Enter**.

Example:

```
Configure SNMP Network Management? [no]:
```

A summary of the available interfaces displays. The interface numbering that appears depends on the type of platform and on the installed interface modules and cards.

Example:

```
Current interface summary
Interface                  IP-Address      OK? Method Status                Prol
Ethernet0/0                unassigned      YES NVRAM  administratively down dow
Ethernet1/0                unassigned      YES NVRAM  administratively down dow
Serial2/0                  unassigned      YES NVRAM  administratively down dow
Serial3/0                  unassigned      YES NVRAM  administratively down dow
Loopback0                  1.1.1.1         YES NVRAM  up                    up
Enter interface name used to connect to the
management network from the above interface summary:
```

Step 10 Select an interface to connect the router to the management network:

Example:

```
Enter interface name used to connect to the
management network from the above interface summary: Ethernet0/0
```

Step 11 Respond to the prompts as appropriate for your network. In this example, IP is configured: an IP address is entered and the current subnet mask is accepted. The screen displays the command script created.

Example:

```
Configuring interface Ethernet0/0:
  Configure IP on this interface? [no]: yes
    IP address for this interface: 172.17.1.1
    Subnet mask for this interface [255.255.0.0] :
    Class B network is 172.17.0.0, 16 subnet bits; mask is /16
The following configuration command script was created:
hostname Router
enable secret 5 $1$1Gg9$GuxXfUUBBfVqGvlW4psIm1
enable password cts54tnl
line vty 0 4
password tls6gato
no snmp-server
!
no ip routing
!
interface Ethernet0/0
no shutdown
ip address 172.17.1.1 255.255.0.0
!
interface Ethernet1/0
shutdown
no ip address
!
interface Serial2/0
shutdown
no ip address
!
interface Serial3/0
shutdown
no ip address
!
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]:
```

Step 12 Enter **2orpressEnter**to save the configuration file to NVRAM and exit.

Example:

```
[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]: 2
```

The screen displays the following:

Example:

```
Building configuration...
[OK]
Use the enabled mode 'configure' command to modify this configuration.
Router#
```

```
00:01:32: %LINK-3-UPDOWN: Interface Ethernet0/0, changed state to up
00:01:33: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/0, changed p
```

What to Do Next

Proceed to the "Verifying the Configuration" section.

Using the System Configuration Dialog to Make Configuration Changes

The *System Configuration Dialog* is an alternative to the CLI when configuration changes do not require the level of detail the CLI provides. For example, you can use the System Configuration Dialog to add a protocol suite, make addressing scheme changes, or configure a newly installed interface. Although you can use configuration modes available through the CLI to make these changes, the *System Configuration Dialog* provides you a high-level view of the configuration and guides you through the configuration process.

Before You Begin

When you add or modify hardware and need to update a configuration, refer to documentation for your platform for details about physical and logical port assignments.

Note The System Configuration Dialog does not allow you to randomly select or enter parameters for configuration. You must move through the dialog step by step until the screen shows the information you want to change.

SUMMARY STEPS

1. **enable**
2. **setup**
3. **Follow Steps 3 through 12 in** the **Detailed Steps in the preceding "Using the System Configuration Dialog to Create an Initial Configuration File" section on page 5 .**
4. Verify the configuration is modified correctly. Refer to the "Verifying the Configuration" section.

DETAILED STEPS

Step 1 **enable**

The **enable** command enters privileged EXEC mode.

Example:

```
Router> enable
Router#
```

Step 2 **setup**

The **setup** command puts the router in **setup** mode.

Example:

```
Router# setup
```

The screen displays the following:

Example:

```
--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]:
```

Enter yes at**thepromptto continuethedialog**.

Example:

```
Continue with configuration dialog? [yes/no]: yes

The screen displays the following:
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]:
```

Step 3 **Follow Steps 3 through 12 in** the **Detailed Steps in the preceding "Using the System Configuration Dialog to Create an Initial Configuration File" section on page 5 .**

Step 4 Verify the configuration is modified correctly. Refer to the "Verifying the Configuration" section.

Verifying the Configuration

Perform this task to verify that the configuration you created using the System Configuration Dialog is operating correctly.

SUMMARY STEPS

1. **show interfaces**
2. **show ip interface brief**
3. **show configuration**

DETAILED STEPS

Step 1 **show interfaces**

This command verifies that the interfaces are operating correctly and that they and the line protocol are in the correct state: up or down.

Step 2 **show ip interface brief**
This command displays a summary status of the interfaces configured for IP.

Step 3 **show configuration**
This command verifies that the correct hostname and password were configured.

Example

This example is the verification of the configuration file created in the "Using the System Configuration Dialog to Create an Initial Configuration File" section.

```
Router# show interfaces
Ethernet0/0 is up, line protocol is up
  Hardware is AmdP2, address is aabb.cc03.6c00 (bia aabb.cc03.6c00)
  Internet address is 172.17.1.1/16
  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 never, output 00:00:06, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  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 input packets with dribble condition detected
     11 packets output, 1648 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
Ethernet1/0 is administratively down, line protocol is down
  Hardware is AmdP2, address is aabb.cc03.6c01 (bia aabb.cc03.6c01)
  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 never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  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 input packets with dribble condition detected
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
Serial2/0 is administratively down, line protocol is down
  Hardware is M4T
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
```

```
  Keepalive set (10 sec)
  Restart-Delay is 0 secs
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1158 kilobits/sec
  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, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions     DCD=up  DSR=up  DTR=down  RTS=down  CTS=up
Serial3/0 is administratively down, line protocol is down
  Hardware is M4T
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, crc 16, loopback not set
  Keepalive set (10 sec)
  Restart-Delay is 0 secs
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/0/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1158 kilobits/sec
  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, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     1 carrier transitions     DCD=down  DSR=down  DTR=up  RTS=up  CTS=down
Loopback0 is up, line protocol is up
  Hardware is Loopback
  Internet address is 1.1.1.1/32
  MTU 1514 bytes, BW 8000000 Kbit, DLY 5000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation LOOPBACK, loopback not set
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  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, 0 interface resets
Router# show ip interface brief
Interface                  IP-Address      OK? Method Status                Prol
Ethernet0/0                172.17.1.1      YES manual up                    up
Ethernet1/0                unassigned      YES manual administratively down dow
Serial2/0                  unassigned      YES manual administratively down dow
Serial3/0                  unassigned      YES manual administratively down dow
Loopback0                  1.1.1.1         YES NVRAM  up                    up
Router# show configuration
Using 1029 out of 8192 bytes
!
version 12.3
service timestamps debug uptime
```

```
service timestamps log uptime
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$1Gg9$GuxXfUUBBfVqGvlW4psIm1
enable password cts54tnl
!
no aaa new-model
!
resource manager
!
clock timezone PST -8
ip subnet-zero
no ip routing
!
!
!
!
!
interface Loopback0
 ip address 1.1.1.1 255.255.255.255
 no ip route-cache
!
interface Ethernet0/0
 ip address 172.17.1.1 255.255.0.0
 no ip route-cache
!
interface Ethernet1/0
 no ip address
 no ip route-cache
 shutdown
!
interface Serial2/0
 no ip address
 no ip route-cache
 shutdown
 serial restart-delay 0
!
interface Serial3/0
 no ip address
 no ip route-cache
 shutdown
 serial restart-delay 0
!
!
ip classless
no ip http server
!
!
!
!
control-plane
!
!
line con 0
 transport preferred all
 transport output all
line aux 0
 transport preferred all
 transport output all
line vty 0 4
 password tls6gato
 login
 transport preferred all
 transport input all
 transport output all
!
end
```

Configuration Examples for Using Cisco IOS Setup Mode to Configure a Cisco Networking Device

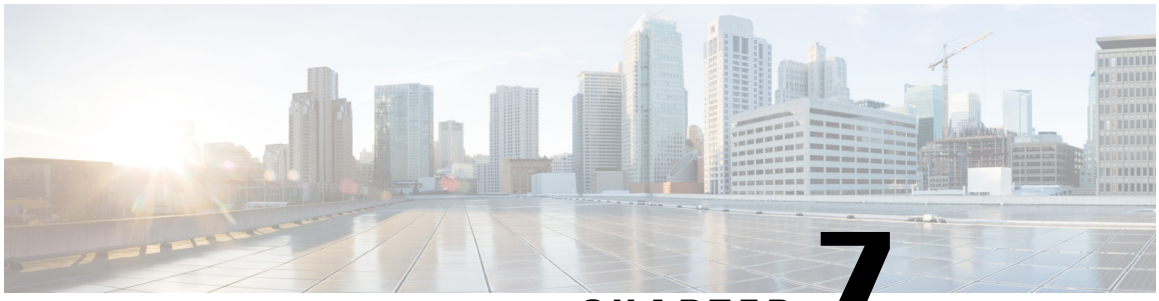
Example Configuring Ethernet Interface 0 Using the System Configuration Dialog

In the following example, the System Configuration Dialog is used to configure Ethernet interface 0 with an IP address.

Note Prompts and the order in which they appear on the screen vary depending on the platform and the interfaces installed in the device.

```
R1# setup
--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: yes
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]: yes
Configuring global parameters:
  Enter host name [R1]: Router
  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: 1g2j3mmc
  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: cts54tnl
  The virtual terminal password is used to protect
  access to the router over a network interface.
  Enter virtual terminal password: tls6gato
  Configure SNMP Network Management? [no]:
Current interface summary
Interface                  IP-Address      OK? Method Status                Prol
Ethernet0/0                172.17.1.1      YES manual up                    up
Ethernet1/0                unassigned      YES manual administratively down dow
Serial2/0                  unassigned      YES manual administratively down dow
Serial3/0                  unassigned      YES manual administratively down dow
Loopback0                  1.1.1.1         YES NVRAM  up                    up
Enter interface name used to connect to the
management network from the above interface summary: Ethernet0/0
Configuring interface Ethernet0/0:
  Configure IP on this interface? [no]: yes
    IP address for this interface: 172.17.1.1
    Subnet mask for this interface [255.255.0.0] :
    Class B network is 172.17.0.0, 16 subnet bits; mask is /16
The following configuration command script was created:
hostname Router
enable secret 5 $1$1Gg9$GuxXfUUBBfVqGvlW4psIm1
enable password cts54tnl
line vty 0 4
password tls6gato
no snmp-server
!
no ip routing
!
interface Ethernet0/0
no shutdown
```

```
ip address 172.17.1.1 255.255.0.0
!
interface Ethernet1/0
shutdown
no ip address
!
interface Serial2/0
shutdown
no ip address
!
interface Serial3/0
shutdown
no ip address
!
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]:
Building configuration...
[OK]
Use the enabled mode 'configure' command to modify this configuration.
Router#
00:01:32: %LINK-3-UPDOWN: Interface Ethernet0/0, changed state to up
00:01:33: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/0, changed p
```



CHAPTER 7

Using AutoInstall to Remotely Configure Cisco Networking Devices

AutoInstall enables remote, automatic configuration of networking devices. AutoInstall is typically used to set up new networking devices remotely. You can, however, use AutoInstall to configure existing networking devices after you remove the configuration file from their NVRAM. The AutoInstall process uses preexisting configuration files that are stored on a TFTP server.

In this module the term networking device means a router that runs Cisco IOS software. Also, the following terms are used interchangeably:

- initial configuration and startup configuration
- *set up* and *configure*

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Using AutoInstall to Remotely Configure Cisco Networking Devices

Services and Servers Used by AutoInstall Dynamic Assignment of IP Addresses

The network must be able to provide the dynamic assignment of an IP address to the networking device that is being configured with AutoInstall. The type of IP address assignment server that is used depends on the type of connection that the networking that is being configured with AutoInstall has to the network.

AutoInstall uses these types of IP address servers:

DHCP Servers

Networking devices using AutoInstall over a LAN connection require a DHCP server to provide an IP address dynamically. This requirement applies to Fast Ethernet, Token Ring, and FDDI interfaces. The network must be configured to provide IP connectivity between the DHCP server and any devices that are using AutoInstall over LAN connections.

DHCP (defined in RFC 2131) is an extension of the functionality provided by the Bootstrap Protocol (defined in RFC 951). DHCP provides the framework for passing configuration information to hosts on a TCP/IP network. DHCP adds the capability of automatic allocation of reusable network addresses and additional configuration options such as a router (gateway) IP address, a TFTP server IP address, the name of a boot file to load, and the domain name to use. DHCP servers can be configured on routers, UNIX servers, Microsoft Windows-based servers, and other platforms.

DHCP servers typically assign IP addresses from a pool of IP addresses randomly. It is possible for a device that uses DHCP to obtain its IP address to have a different IP address every time it is connected to the network. This creates a problem for the AutoInstall process when you want to ensure that a particular device is assigned a specific hostname during the AutoInstall process. For example, if you are installing routers on different floors in a remote site and each router is supposed to be assigned a name that indicates its location, such as **ChicagoHQ-1st** and **ChicagoHQ-2nd**, you need to ensure that each device gets the IP address that will be mapped to its correct hostname.

The process of ensuring that a device is assigned a specific IP address is referred to as *creating a reservation*. A reservation is a manually configured relationship between an IP address and a physical layer address of a LAN interface on the device. Many Cisco IOS XE-based devices do not use their MAC address when they request an IP address via DHCP. They use a much longer client identifier instead. Due to the complexity of identifying the client identifier so that you can preconfigure a reservation, and the complexity of finding out if the new device uses its MAC address or the client identifier, we recommend that you allow a new device to obtain an IP address without using a DHCP reservation first in order to discover if the device is using its MAC address or a client identifier. When you have learned how the new device is identifying itself to the DHCP server, you can make a note of the format and create a reservation for it. The next time the new device is rebooted it should obtain the IP address that you reserved to ensure that the new device is assigned the correct hostname. Refer to the information on creating DHCP reservations that was provided with your DHCP server software. The process for creating reservations using Cisco IOS XE based DHCP servers is explained in the Using AutoInstall to Set Up Devices Connected to LANs: Example module. This section includes instructions for identifying the client identifier before the device is connected to the network so that you can preconfigure the DHCP reservations.

Note This document uses a Cisco router as the DHCP server for using AutoInstall to configure LAN-connected networking devices. If you are using a different device as your DHCP server ensure that you have the user documentation for it available in the event that you need help configuring it.

Note There are several configuration parameters such as TFTP server addresses, DNS server addresses, domain names and so on, that can be provided to LAN-connected clients by DHCP servers during the process of assigning IP addresses to clients. These parameters are not required by AutoInstall, therefore they are not included in this document. If you know how to use these parameters you can include them in your DHCP server configuration when you are using AutoInstall to setup your networking devices.

For more information on DHCP services visit the IETF RFC site (http://www.ietf.org/rfc.html) and look for RFCs about DHCP. Most server operating systems support DHCP servers. Refer to the documentation that was provided with your operating system for more information.

SLARP Servers

A router that is being configured with AutoInstall over a serial interface using HDLC encapsulation will send a Serial Line ARP (SLARP) request for an IP address over the serial interface that is connected to the staging router.

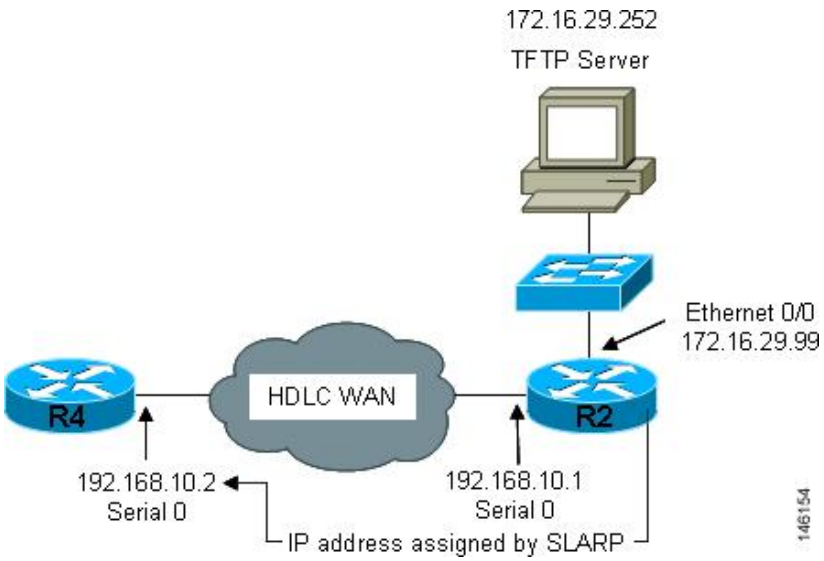
The serial interface of the staging router must be configured with an IP address in which the host portion is 1 or 2, such as 192.168.10.1 or 192.168.10.2. The staging router will send a SLARP response to the router that is being configured with AutoInstall that contains the value that the staging router is not using. For example, if the interface on the staging router that is connected to the router that is being configured with AutoInstall is using 192.168.10.1 as its IP address, the staging router will send a SLARP response with a value of 192.168.10.2 to the router that is being configured with AutoInstall.

Tip If you are using a mask of 255.255.255.252 on the serial interface of the staging router SLARP will assign the available IP host address to the new device. For example, if you assign IP address 198.162.10.5 255.255.255.252 to serial 0 on the staging router, SLARP will assign 198.162.10.6 to the new device. If you assign IP addresses 198.162.10.6 255.255.255.252 to serial 0 on the staging router SLARP will assign 198.162.10.5 to the new device.

The figure below shows an example of SLARP.

In the figure below, the IP address of serial interface 0 on the staging router (R2) is 192.168.10.1. SLARP therefore assigns the IP address 192.168.10.2 to serial interface 0 on the new device.

Figure 1: Using SLARP to Assign an IP Address to a New Device



Note AutoInstall over a serial interface using HDLC can be performed only over the first serial port on a new device (serial interface 0 or serial interface x/0). The staging router and new device must be directly connected using the first serial interface port on the new device; for example, serial 0/0 or if the first serial port is in the second slot of the device, serial 2/0.

Tip The IP address that is assigned to the router that is being configured with AutoInstall by SLARP from the staging router is the IP address that you must use in the **ip host** *hostname ip-address*command in the AutoInstall network-confg or cisconet.cfg file to ensure that the router that is being configured with AutoInstall is assigned the correct hostname so that it can request its host-specific configuration file.

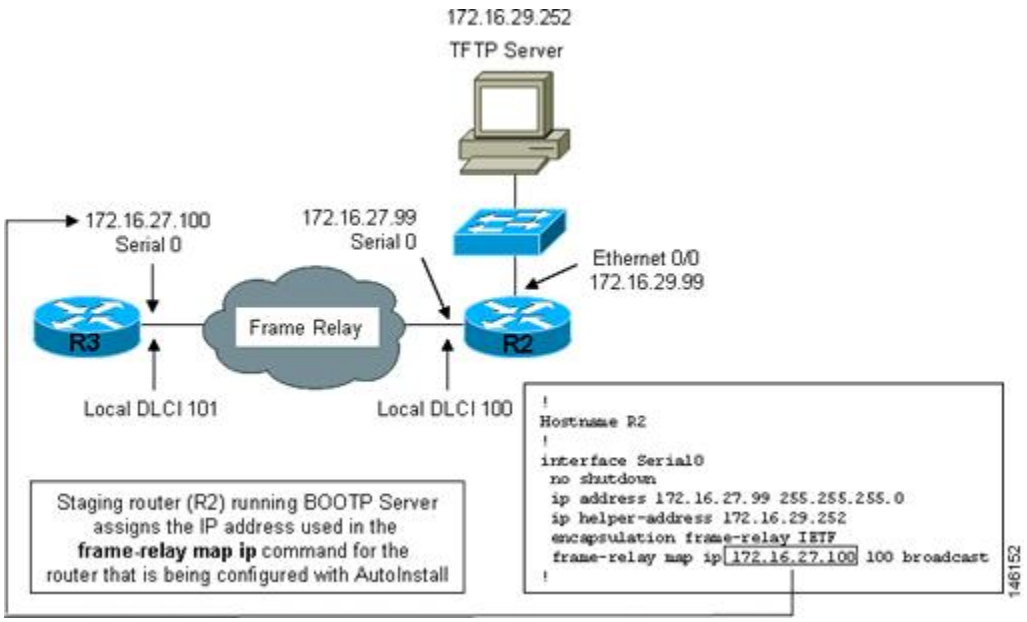
BOOTP Servers

A router that is being configured with AutoInstall over a serial interface using Frame Relay encapsulation will send a BOOTP request for an IP address over the serial interface that is connected to the staging router.

The staging router learns the correct IP address to provide in its BOOTP response to the router that is being configured with AutoInstall by examining the **frame-relay map ip** *ip-address dlci* command that is configured on the interface that it is using to connect to the router that is being configured with AutoInstall.

In the figure below R2 is the staging router. R2 has the **frame-relay map ip 172.16.27.100 100** broadcast command configured on interface serial 0. When R2 receives the BOOTP request for an IP address from R3 during the AutoInstall process, R3 will reply with 172.16.27.100.

Figure 2: Example of Using BOOTP for Autoinstall Over a Frame Relay Network



Tip The limitation imposed by SLARP in which the IP addresses for the new device and the staging router must end in either .1 or .2 does not apply to BOOTP. BOOTP for AutoInstall over Frame Relay supports all host addresses for the IP address subnet that is assigned to the Frame Relay circuit between the router that is being configured with AutoInstall and the staging router.

Tip The IP address that is assigned to the router that is being configured with AutoInstall by BOOTP from the staging router is the IP address that you must use in the **ip host** *hostname ip-address*command in the AutoInstall network-confg or cisconet.cfg file to ensure that the router that is being configured with AutoInstall is assigned the correct hostname so that it can request its host-specific configuration file.

Note AutoInstall over a serial interface using Frame Relay encapsulation can be performed only over the first serial port on a new device (serial interface 0 or serial interface x/0). The staging router and new device must be directly connected using the first serial interface port on the new device; for example, serial 0/0 or if the first serial port is in the second slot of the device, serial 2/0.

Services and Servers Used by AutoInstall IP-to-Hostname Mapping

If you want the networking device to load a full configuration file during the AutoInstall process, the networking device must be able to determine its hostname so that it can request the configuration file that you created specifically for it.

The following caveats apply to the provisioning of IP address to hostname mapping for AutoInstall:

- Any networking device that is being configured with AutoInstall can determine its hostname by loading one of the AutoInstall network configuration files (network-confg or cisconet.cfg) from the TFTP server that contain the **iphost***hostnameip-address* commands. For example, to map host R3 to IP address 198.162.100.3, the network-confg or cisconet.cfg file must contain the **iphostr3198.162.100.3** command.
- A networking device that is being configured with AutoInstall over a LAN interface can also determine its hostname by querying a DNS server. If the DNS server is not connected to the same LAN the device must learn the IP address of the DNS server from the DHCP server during the process of obtaining its dynamically assigned IP address from the DHCP server.

DNS Servers

DNS servers are used to provide a network service that maps hostnames to IP addresses and IP addresses to hostnames (reverse DNS lookups). Anytime that you use a hostname to initiate an IP connection to a host, your PC must determine the IP address that is assigned to the hostname that you want to contact. For example, when you visit Cisco's website (http://www.cisco.com/) your PC sends a DNS query to a DNS server to discover the current IP address that can be used to contact Cisco's website.

For more information on DNS services visit the IETF RFC site (http://www.ietf.org/rfc.html) and look for RFCs about DNS. The Name Server LookUp tool (nslookup) is very useful for learning more about DNS. There are several excellent websites available about nslookup that you can find by searching for them.

Services and Servers Used by AutoInstall Storage and Transmission of Configuration Files

TFTP is a protocol used to transfer files between devices on a network. A TFTP server is a device that uses TFTP to transfer files to devices. TFTP servers can be configured on UNIX servers, Microsoft Windows-based PCs and servers, and other platforms.

Tip If you do not have a TFTP server available you can configure a Cisco IOS-based router as a TFTP server using the **tftp-serverfile-system**:*filename* command. Refer to the Configuring Basic File Transfer Services feature for more information on configuring your router as a TFTP server.

Cisco routers use TFTP to load the configuration files that are required for AutoInstall. You must have a TFTP server deployed in your network to provide file storage and file transmission services to the devices that will be using AutoInstall.

For more information on TFTP services visit the IETF RFC site (http://www.ietf.org/rfc.html) and look for RFCs about TFTP. There are several excellent websites available about TFTP that you can find by searching for them. Several freeware and shareware versions of TFTP servers for various operating systems and hardware platforms are available from the Internet.

The following caveats apply to the provisioning of TFTP servers for AutoInstall:

- Devices using AutoInstall over a LAN--If the TFTP server and the devices using AutoInstall are on different LAN segments, you must either configure the **iphelper-address** *address* command on all of the interfaces that will receive TFTP session initialization requests from the devices that are using AutoInstall.
- Devices using AutoInstall over a WAN--If the devices using AutoInstall are connected to a WAN, you must configure the **iphelper-address** *address* command on all of the interfaces that will receive TFTP session initialization requests from devices that are using AutoInstall.

ip helper-address

If the new device does not learn the IP address of the TFTP server via DHCP option 150, it will transmit the TFTP session initialization requests as network layer broadcasts using the IP destination broadcast address of 255.255.255.255. Routers block network layer broadcast datagrams which prevents the TFTP session initialization requests from reaching the TFTP server, and AutoInstall will fail. The solution to this problem is to use the **ip helper-address** *address* command. The **ip helper-address** *address* command changes the broadcast address of TFTP session initialization request from 255.255.255.255 to the address that is configured with the *address* argument. For example, the **ip helper-address 172.16.29.252** command will change IP destination broadcast address of 255.255.255.255 to 172.16.29.252.

Networking Devices Used by AutoInstall

Device That Is Being Configured with AutoInstall

A device that is being configured with AutoInstall can be any Cisco IOS XE-based router that supports AutoInstall and does not have a configuration file in its NVRAM.

Staging Router

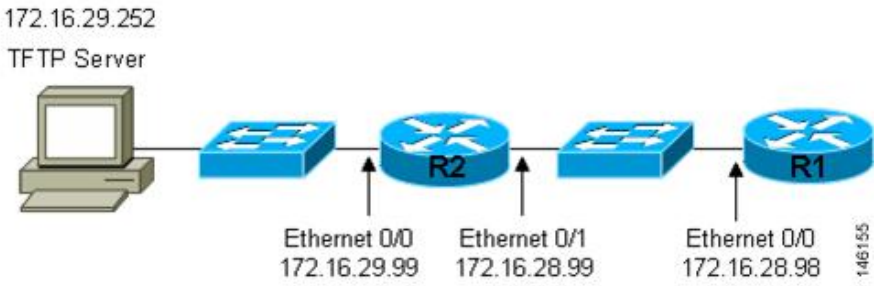
A staging router acts as an intermediary between the TFTP server (to which it must have IP connectivity) and a device that is being configured with AutoInstall when the new device and the TFTP server are connected to different networks. In the figure below R1 requires a staging router because it is connected to a different LAN segment than the TFTP server.

Staging routers are required in the following situations:

- Devices using AutoInstall over a LAN--If the TFTP and/or DHCP servers and the devices using AutoInstall are on different LAN segments you must use a staging router.

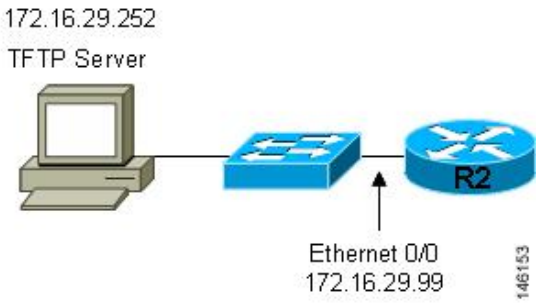
- Devices using AutoInstall over a WAN--If the devices using AutoInstall are connected to a WAN, you must configure the **ip helper-address** *address* command on all of the directly connected interfaces that will receive TFTP session initialization requests from the devices that are using AutoInstall.

Figure 3: Example of AutoInstall That Requires a Staging Router



Staging routers are not required when the new device that is being configured with AutoInstall is connected to the same LAN segment as the TFTP and DHCP servers. In the figure below R2 does not require a staging server to use AutoInstall because it is on the same LAN segment as the TFTP server.

Figure 4: Example of AutoInstall That Does Not Require a Staging Router



Intermediate Frame Relay-ATM Switching Device

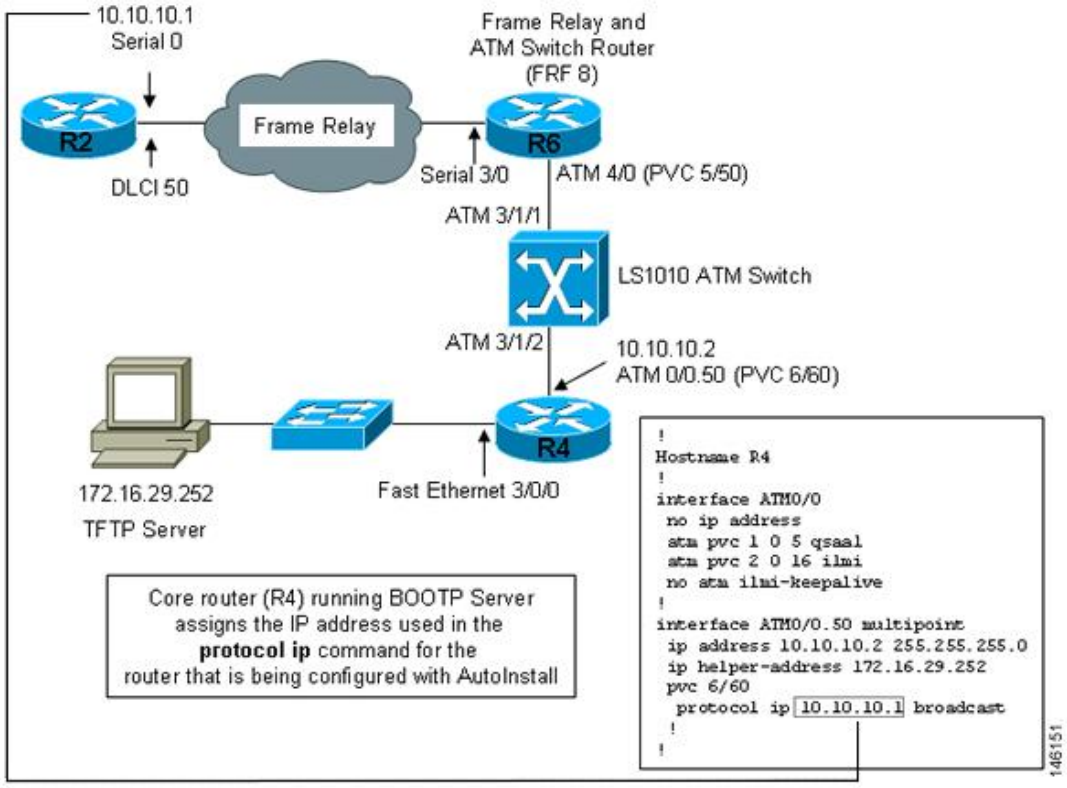
An intermediate Frame Relay-ATM switching device is one that can perform both routing and switching operations. Frame Relay-ATM switching devices are used to connect Frame Relay and ATM networks.

The AutoInstall over Frame Relay-ATM Interworking Connections feature modifies the AutoInstall process to use Frame Relay encapsulation defined by the IETF standard instead of the Frame Relay encapsulation defined by Cisco.

The figure below shows an example topology using AutoInstall over Frame Relay-ATM Interworking Connections. Router R6 does the Frame Relay to ATM Service Internetworking (FRF8) conversion for Frame

Relay DLCI 50 to ATM VPI/VCI 5/50. The LS1010 switch routes the VPI/VCI combination used by R6 (5/50) to the VPI/VCI combination used by R4 (6/60).

Figure 5: Example Topology for AutoInstall over Frame Relay-ATM Interworking Connections



Configuration Options for AutoInstall

You can provision your network to support AutoInstall using several different combinations of devices and services. For example:

- You can provision all of the services required for AutoInstall (except dynamic IP address assignment using SLARP or BOOTP that must be preformed by a Cisco router) on one network server, or you can provision each service on a different network server.
- You can provision the DHCP service on a Cisco router.
- The device using AutoInstall can determine its IP address from a DNS server, or you can use one of the AutoInstall network configuration files (network-confg or cisconet.cfg) that contain the **ip host** *hostname ip-address* commands.
- You can use provision AutoInstall to load a full configuration or a partial configuration onto a device that is using AutoInstall.

This module focuses on some of the most common methods for provisioning AutoInstall. Refer to the How to Use AutoInstall to Remotely Configure Cisco Networking Devices module for information on the most common methods for provisioning AutoInstall.

The AutoInstall Process

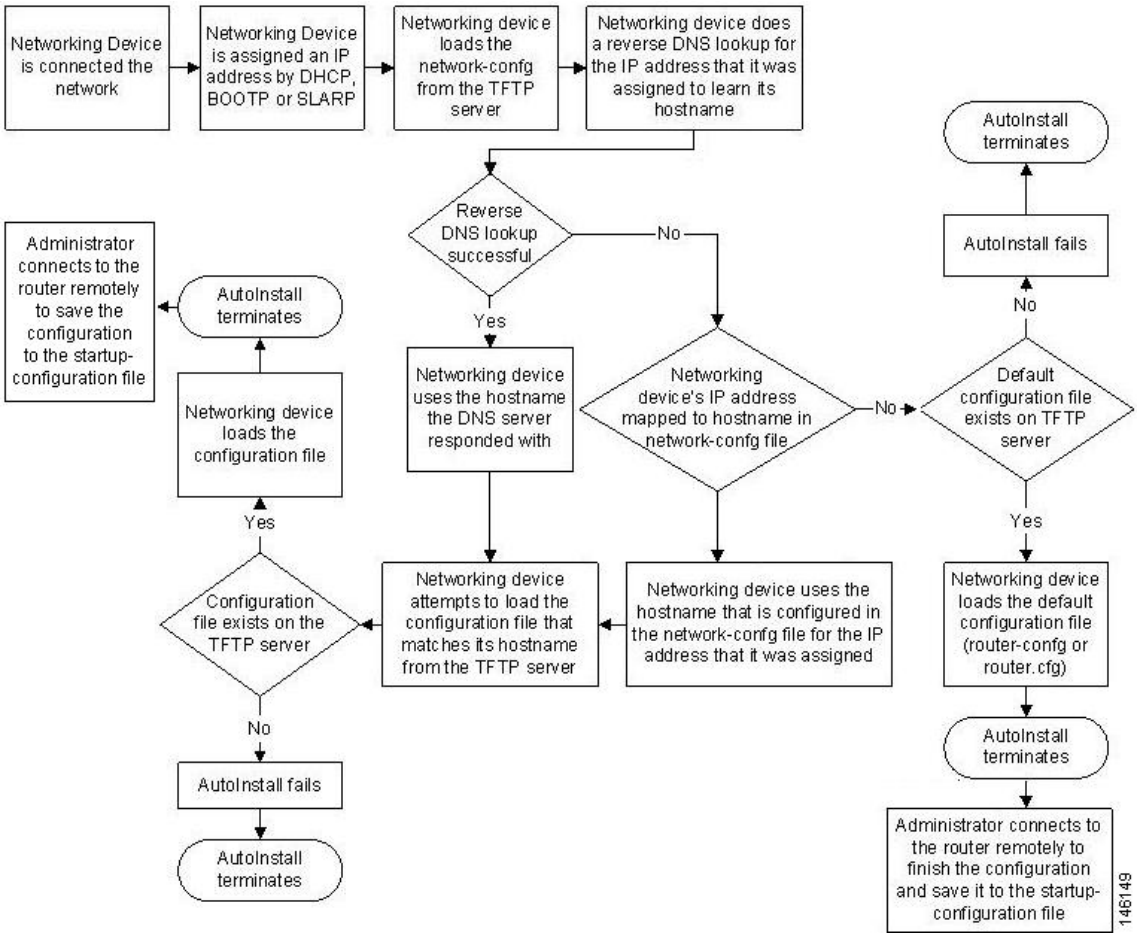
The AutoInstall process begins when a networking device that does not have any files in its NVRAM is connected to the network.

Timesaver

You can decrease the time that the AutoInstall process takes to complete by only connecting the interface on the networking device that you want to use for AutoInstall until the AutoInstall process has finished. For example, if you want the networking device to perform AutoInstall over a WAN interface and you connect its LAN interfaces and its WAN interfaces the networking device will attempt to perform AutoInstall over the LAN interfaces before it attempts to use the WAN interfaces. Leaving the LAN interfaces disconnected until the AutoInstall process is finished causes the networking device to initiate the AutoInstall process over its WAN interface immediately.

The following figure shows the basic flow of the AutoInstall process using the configuration files.

Figure 6: AutoInstall Process Flowchart (Using Configuration Files)



How to Use AutoInstall to Remotely Configure Cisco Networking Devices

This section describes the how to prepare a router for AutoInstall. Additional examples for using AutoInstall for new routers connected to LANs, HDLC WANs, and Frame Relay networks, are provided in the Configuration Examples for Using AutoInstall to Remotely Configure Cisco Networking Devices module.

In most cases you need to configure a staging router through which a new device running AutoInstall sends TFTP, BOOTP, and DNS requests.

Tip In all cases, you must verify and save the configuration on the networking device after the AutoInstall process is complete. If you do not save the configuration, you must repeat the entire process.

Disabling the SDM Default Configuration File

Perform this task if SDM was preinstalled on your device and you want to use Setup to build an initial configuration file. SDM remains on the device.

Perform this task if SDM was pre installed on your device and you want to use AutoInstall to configure the device instead. SDM remains on the device.

SUMMARY STEPS

1. Connect the console cable, shipped with your device, from the console port on the device to a serial port on your PC. Refer to the hardware installation guide for the device for instructions.
2. Connect the power supply to the device, plug the power supply into a power outlet, and turn on the device. Refer to the quick start guide for the device for instructions.
3. Use Hyperterminal or a similar terminal emulation program on your PC, with the following terminal emulation settings, to connect to the device:
4. **enable**
5. **erase startup-config**
6. **reload**

DETAILED STEPS

Step 1 Connect the console cable, shipped with your device, from the console port on the device to a serial port on your PC. Refer to the hardware installation guide for the device for instructions.

Step 2 Connect the power supply to the device, plug the power supply into a power outlet, and turn on the device. Refer to the quick start guide for the device for instructions.

Step 3 Use Hyperterminal or a similar terminal emulation program on your PC, with the following terminal emulation settings, to connect to the device:

- 9600 baud

- 8 data bits, no parity, 1 stop bit
- No flow control

Step 4 **enable**
Enter privileged EXEC mode.

enable

Example:

```
Router> enable
Router#
```

Step 5 **erase startup-config**
Erases the existing configuration in NVRAM.

Example:

```
Router# erase startup-config
```

Step 6 **reload**
Initiates the reload process. The router will initiate the AutoInstall process after it finishes the reload process.

Example:

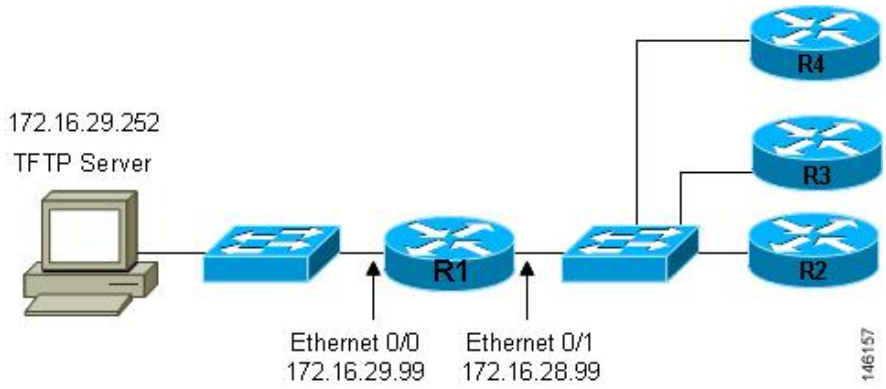
```
Router# reload
```

Configuration Examples for Using AutoInstall to Remotely Configure Cisco Networking Devices

Using AutoInstall to Set Up Devices Connected to LANs Example

This task uses the network in the figure below. This task will show how to use AutoInstall to setup routers R2, R3, and R4. Router R1 is the DHCP server that will be used to assign the IP address for Fast Ethernet 0/0 on the new routers during the AutoInstall process.

Figure 7: Network Topology for Assigning AutoInstall Configuration Files For Specific Devices



Every DHCP client has a unique DHCP client identifier. The DHCP client identifier is used by DHCP servers to keep track of IP address leases and for configuring IP address reservations. You need to know the DHCP client identifier for each of the networking devices that you want to configure with AutoInstall so that you can configure the DHCP IP address reservations which will ensure that each device is provided with the correct IP address, and subsequently its unique configuration file. You can determine the DHCP client identifier manually or automatically.

To use AutoInstall to setup routers R2, R3, and R4, perform following tasks:

Determining the Value for the DHCP Client Identifier Manually

If you want to determine the value for the client identifiers automatically, you do not need to perform this task. Proceed to the Determining the Value for the DHCP Client Identifier Automatically module.

You must know the MAC address of the Fast Ethernet interface that will be used to connect the router to the LAN during the AutoInstall process to determine the client identifier manually. This requires connecting a terminal to the router, and powering it on, so that you can enter the **show interface** *interface-type interface-number* command.

The client-identifier looks like this:

0063.6973.636f.2d30.3030.362e.3533.6237.2e38.6537.312d.4661.332f.30

The format is *nullcisco-0006.53b7.8e71-fa3/0* where *0006.53b7.8e71* is the MAC address and *fa3/0* is the short interface name for the interface that the IP address request is made for.

The values for the short-if-name field can be obtained from an SNMP workstation with the Cisco MIBs installed. This is an example of how to map ifIndex to an interface on Cisco IOS:

```
snmpwalk -c public ponch ifName
IF-MIB::ifName.1 = STRING: AT2/0
IF-MIB::ifName.2 = STRING: Et0/0
IF-MIB::ifName.3 = STRING: Se0/0
IF-MIB::ifName.4 = STRING: BR0/0
```

Use the **show interface** *interface-type interface-number* command to display the information and statistics for a Fast Ethernet interface.

```
R6> show interface fastethernet 3/0
FastEthernet3/0 is up, line protocol is up
  Hardware is AmdFE, address is 0006.53b7.8e71 (bia 0006.53b7.8e71)
.
.
.
R6>
```

The MAC address for FastEthernet 3/0 on R6 is 0006.53b7.8e71. The format of the client identifier for this interface is nullcisco-0006.53b7.8e71-fa3/0.

Note The short interface name for Fast Ethernet interfaces is fa.

The table below shows the values for converting characters to their hexadecimal equivalents. The last row in the second table below shows the client identifier for Fast Ethernet 3/0 on R6 (nullcisco-0006.53b7.8e71-fa3/0).

Table 7: Hexadecimal to Character Conversion Chart

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
00	NUL	1a	SUB	34	4	4e	N	68	h
01	SOH	1b	ESC	35	5	4f	O	69	I
02	STX	1c	FS	36	6	50	P	6a	j
03	ETX	1d	GS	37	7	51	Q	6b	k
04	EOT	1e	RS	38	8	52	R	6c	l
05	ENQ	1f	US	39	9	53	S	6d	m
06	ACK	20		3a	:	54	T	6e	n
07	BEL	21	!	3b	;	55	U	6f	o
08	BS	22	"	3c	<	56	V	70	p
09	TAB	23	#	3d	=	57	W	71	q
0A	LF	24	$	3e	>	58	X	72	r
0B	VT	25	%	3f	?	59	Y	73	s

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
0C	FF	26	&	40	@	5a	Z	74	t
0D	CR	27	‘	41	A	5b	[	75	u
0E	SO	28	(	42	B	5c	\	76	v
0F	SI	29	)	43	C	5d	]	77	w
10	DLE	2a	*	44	D	5e	^	78	x
11	DC1	2b	+	45	E	5f	_	79	y
12	DC2	2c	,	46	F	60	`	7a	z
13	DC3	2d	-	47	G	61	a	7b	{
14	DC4	2e	.	48	H	62	b	7c	|
15	NAK	2f	/	49	I	63	c	7D	}
16	SYN	30	0	4a	J	64	d	7e	~
17	ETB	31	1	4b	K	65	e	7f	D
18	CAN	32	2	4c	L	66	f		
19	EM	33	3	4d	M	67	g		

Table 8: Conversion of nullcisco-0006.53b7.8e71-fa3/0 To A Client Identifier

00	c	i	s	c	o	-	0	0	0	6	.	5	3	b	7	.	8	e	7	1	-	f	a	3	/	0
00	63	69	73	63	6f	2d	30	30	30	36	2e	35	33	62	37	2e	38	65	37	31	2d	46	61	33	2f	30

R4

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R4.

```
R4> show interface FastEthernet 0/0
FastEthernet0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb0e (bia 00e0.1eb8.eb0e)
```

The MAC address for Fast Ethernet 0/0 on R4 is 00e0.1eb8.eb0e. The format of the client identifier for this interface is nullcisco-00e0.1eb8.eb0e-et0.

Note The short interface name for Fast Ethernet interfaces is et.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R4 is shown in the last row of the table below.

Table 9: Conversion of null.cisco-00e0.1eb8.eb0e-et0 To A Client Identifier for R4

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	0	e	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	30	65	2d	45	74	30

R3

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R3.

```
R3> show interface FastEthernet 0/0
FastEthernet0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb73 (bia 00e0.1eb8.eb73)
```

The MAC address for Fast Ethernet 0/0 on R3 is 00e0.1eb8.eb73. The format of the client identifier for this interface is: nullcisco-00e0.1eb8.eb73-et0.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R3 is shown in the last row of the table below.

Table 10: Conversion of null.cisco-00e0.1eb8.eb73-et0 To A Client Identifier for R3

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	7	3	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	37	33	2d	45	74	30

R2

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R2.

```
R2> show interface Fast Ethernet 0/0
FastEthernet0/0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb09 (bia 00e0.1eb8.eb09)
```

The MAC address for Fast Ethernet 0/0 on R2 is 00e0.1eb8.eb09. The format of the client identifier for this interface is nullcisco-00e0.1eb8.eb09-et0.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R2 is shown in the last row of the table below

Table 11: Conversion of null.cisco-00e0.1eb8.eb09-et0 To A Client Identifier for R2

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	0	9	-	e	t	0

00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	30	39	2d	45	74	30

You have now determined the values for the client identifiers on each router. The final step is to add a period after each group of four characters working from the left to the right as shown below:

- R4-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30
- R3-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30
- R2-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30

Determining the Value for the DHCP Client Identifier Automatically

If you determined the value for the client identifiers manually, you do not need to perform this task. Proceed to the Creating a Private DHCP Pool for Each of The Routers module.

This task will create a DHCP server on R1 that will provide only one IP address. This IP address will used by each new router in sequence while you determine the value of the router's client identifier. By limiting the IP address scope to a single IP address you avoid any possible confusion about which router you are working on. If somebody powers up another router that attempts to start the AutoInstall process, it will not be able to obtain an IP address.

Tip Do not place the network-confg or router configuration files (r4-confg, r3-confg, or r2-confg) in the root directory of the TFTP server yet. You do not want any of the routers to load these files until you have ensured that each router will obtain the correct IP address from the DHCP server so that the router will load the correct configuration file.

This task is broken down into sub-tasks to make it easier to follow (all sub-tasks are required):

Configuring IP on the Interfaces on R1

Configure IP addresses on the Fast Ethernet interfaces. Configure the **ip helper-address** *ip-address* command on Fast Ethernet 0/1.

```
!
interface FastEthernet0/0
 ip address 172.16.29.99 255.255.255.0
!
interface FastEthernet0/1
 ip address 172.16.28.99 255.255.255.0
 ip helper-address 172.16.29.252
!
```

Configuring a DHCP Pool on R1

Configure these commands to setup the temporary DHCP server on R1.

Note This should be the only DHCP server in operation on R1. This should be the only DHCP server that is accessible by the routers that you will be using AutoInstall to setup.

```
!
ip dhcp pool get-client-id
   network 172.16.28.0 255.255.255.0
!
```

Excluding All But One of the IP Addresses from the DHCP Pool on R1

You need to ensure that there is only one IP address available from the DHCP server at any time. Configure the following command to exclude every IP address except 172.16.28.1 from the DHCP pool.

```
!
ip dhcp excluded-address 172.16.28.2 172.16.28.255
!
```

Verifying The Configuration on R1

Verify that the configuration file for R1 has a DHCP server pool configured to provide a single IP address (172.16.28.1) to a DHCP client.

Verify that the configuration file has the IP addresses for the Fast Ethernet interfaces and the **ip helper-address** *ip-address* command.

```
!
ip dhcp excluded-address 172.16.28.2 172.16.28.255
!
ip dhcp pool get-client-id
   network 172.16.28.0 255.255.255.0
!
interface FastEthernet0/0
 ip address 172.16.29.99 255.255.255.0
!
interface FastEthernet0/1
 ip address 172.16.28.99 255.255.255.0
 ip helper-address 172.16.29.252
!
```

Enabling debug ip dhcp server events on R1

You use the display output from the **debug ip dhcp server events** command on the terminal connected to R1 to identify the value of the client identifier for each router.

Enable the **debug ip dhcp server events** command on R1.

```
R1# debug ip dhcp server events
```

Identifying the Value for the Client Identifier on Each of the Routers

This step is repeated for each of the routers. You should only have one of the routers powered-on at any time. When you have identified the value of the client identifier field for the router, you will turn the router off and proceed to the next router.

R4

Connect R4 to the Fast Ethernet network and power it on. The following message will be displayed on the terminal connected to R1 when R4 is assigned the IP address 172.16.28.1.

```
DHCPD: assigned IP address 172.16.28.1 to client
0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30.
```

Copy the client identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30 to a text file and save it. Keep the text file open for the next two routers.

Turn off R4

Release the IP address binding for R4 from the DHCP pool on R1 using the **clear ip dhcp binding *** command on R1.

```
R1# clear ip dhcp binding *
R1#
01:16:11: DHCPD: returned 172.16.28.1 to address pool get-client-id.
```

R3

Connect R3 to the Fast Ethernet network and power it on. The following message will be displayed on the terminal connected to R1 when R3 is assigned the IP address 172.16.28.1.

```
DHCPD: assigned IP address 172.16.28.1 to client
0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30.
```

Copy the client identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30 to the text file and save it. Keep the text file open for the final router.

Turn off R3.

Release the IP address binding for R3 from the DHCP pool on R1 using the **clear ip dhcp binding *** command on R1.

```
R1# clear ip dhcp binding *
R1#
01:16:11: DHCPD: returned 172.16.28.1 to address pool get-client-id.
```

R2

Connect R2 to the Fast Ethernet network and power it on. The following message will be displayed on the terminal connected to R1 when R2 is assigned the IP address 172.16.28.1.

```
DHCPD: assigned IP address 172.16.28.1 to client
0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30.
```

Copy the client identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30 to the text file and save it.

Turn off R2

Release the IP address binding for R2 from the DHCP pool on R1 using the **clear ip dhcp binding *** command on R1.

```
R1# clear ip dhcp binding *
R1#
01:16:11: DHCPD: returned 172.16.28.1 to address pool get-client-id.
```

Client Identifiers for R4, R3, and R2

You have determined the values for the client identifiers on each router.

- R4-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30
- R3-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30
- R2-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30

Removing the DHCP Pool on R1 for Network 172.16.28.0 24

The temporary DHCP pool on the router is no longer required, and must be removed.

```
R1(config)# no ip dhcp pool get-client-id
```

Removing the Excluded Address Range From R1

The command for excluding all of the IP addresses except 172.16.28.1 from the DHCP pool on the router is no longer required, and must be removed.

R1(config)# **no ip dhcp excluded-address 172.16.28.2 172.16.28.255**

Creating a Private DHCP Pool for Each of The Routers

You need to create the private DHCP address pools for each router to ensure that each router is assigned the IP address that maps to its host name in the network-conf file.

```
!
ip dhcp pool r4
   host 172.16.28.100 255.255.255.0
   client-identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30
!
ip dhcp pool r3
   host 172.16.28.101 255.255.255.0
   client-identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30
!
ip dhcp pool r2
   host 172.16.28.102 255.255.255.0
   client-identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30
```

Creating Configuration Files for Each Router

Create the configuration files for each router and place them in the root directory of the TFTP server.

Tip You must include the commands for configuring passwords for remote Telnet access and access to privileged EXEC mode if you are going to access the routers remotely to save their configuration files to NVRAM.

r2-confg

```
!
hostname R2
!
enable secret 7gD2A0
!
interface FastEthernet0/0
 ip address 172.16.28.102 255.255.255.0
```

```
!
interface Serial0/0
 ip address 192.168.100.1 255.255.255.252
 no shutdown
!
interface Serial0/1
 ip address 192.168.100.5 255.255.255.252
 no shutdown
!
no ip http server
ip classless
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0
!
line vty 0 4
 password 5Rf1k9
 login
!
end
```

r3-confg

```
!
hostname R3
!
enable secret 7gD2A0
!
interface FastEthernet0/0
 ip address 172.16.28.101 255.255.255.0
!
interface Serial0/0
 ip address 192.168.100.9 255.255.255.252
 no shutdown
!
interface Serial0/1
 ip address 192.168.100.13 255.255.255.252
 no shutdown
!
no ip http server
ip classless
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 FastEthernet0
!
line vty 0 4
 password 5Rf1k9
 login
!
end
```

r4-confg

```
!
hostname R3
!
enable secret 7gD2A0
!
interface FastEthernet0/0
 ip address 172.16.28.101 255.255.255.0
!
interface Serial0/0
 ip address 192.168.100.9 255.255.255.252
 no shutdown
!
interface Serial0/1
 ip address 192.168.100.13 255.255.255.252
 no shutdown
!
no ip http server
ip classless
```

```
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0
!
line vty 0 4
 password 5Rf1k9
 login
!
end
```

Creating the network-confg file

Create the network-confg file with the **ip host** *hostname ip-address* commands that map the IP addresses that you will be assigning with the DHCP server to the hostname.

```
ip host r4 172.16.28.100
ip host r3 172.16.28.101
ip host r2 172.16.28.102
```

Setting Up the Routers with AutoInstall

You are now ready to set up the three routers (R4, R3, and R2) using AutoInstall.

Connect a terminal to the routers if you want to monitor the progress of AutoInstall. Use Hyperterminal or a similar terminal emulation program on your PC, with the following terminal emulation settings, to connect to the device:

- 9600 baud
- 8 data bits, no parity, 1 stop bit
- No flow control

You should have the following files in the root directory of the TFTP server.

- network-confg
- r4-confg
- r3-confg
- r2-confg

The TFTP server must be running.

Power on each router.

Timesaver You can set up all three routers concurrently.

R4

The following is an excerpt of the messages that are displayed on R4's console terminal during the AutoInstall process:

```
Loading network-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 76 bytes]
Configuration mapped ip address 172.16.28.100 to r4
```

```
Loading r4-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 687 bytes]
```

R3

The following is an excerpt of the messages that are displayed on R3's console terminal during the AutoInstall process:

```
Loading network-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 76 bytes]
Configuration mapped ip address 172.16.28.101 to r3
Loading r3-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 687 bytes]
```

R2

The following is an excerpt of the messages that are displayed on R2's console terminal during the AutoInstall process:

```
Loading network-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 76 bytes]
Configuration mapped ip address 172.16.28.102 to r2
Loading r2-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 687 bytes]
```

TFTP Server Log

The TFTP server log should contain messages similar to the following text.

```
Sent network-confg to (172.16.28.100), 76 bytes
Sent r4-confg to (172.16.28.100),687 bytes
Sent network-confg to (172.16.28.101), 76 bytes
Sent r3-confg to (172.16.28.101),687 bytes
Sent network-confg to (172.16.28.102), 76 bytes
Sent r2-confg to (172.16.28.102),687 bytes
```

Saving the Configuration Files on The Routers

You must save the running configurations on each router to the startup configuration to ensure that the routers retain their configurations if they are ever power cycled.

R4

```
R1# telnet 172.16.28.100
Trying 172.16.28.100 ... Open
User Access Verification
Password:
R4> enable
Password:
R4# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R4# exit
[Connection to 172.16.28.100 closed by foreign host]
R1#
```

R3

```
R1# telnet 172.16.28.101
```

```
Trying 172.16.28.101 ... Open
User Access Verification
Password:
R3> enable
Password:
R3# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R3# exit
[Connection to 172.16.28.101 closed by foreign host]
R1#
```

R2

```
R1# telnet 172.16.28.102
Trying 172.16.28.102 ... Open
User Access Verification
Password:
R2> enable
Password:
R2# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R2# exit
[Connection to 172.16.28.102 closed by foreign host]
R1#
```

Removing the Private DHCP Address Pools from R1

The final step in the AutoInstall process is to remove the private DHCP address pools from R1.

```
R1(config)# no ip dhcp pool r4
R1(config)# no ip dhcp pool r3
R1(config)# no ip dhcp pool r2
```

This is the final task, and step for Using AutoInstall to Setup Devices Connected to LANs.

Additional References

This section provides references related to the basic configuration of a Cisco networking device.

Related Documents

Related Topic	Document Title
Configuring a networking device for the first time using the Cisco IOS XE software feature AutoInstall.	Using AutoInstall to Remotely Configure Cisco Networking Devices
Configuring a networking device using Cisco IOS XE Setup mode	Using Setup Mode to Configure a Cisco Networking Device
Configuration fundamentals and associated commands	*Cisco IOS XE Configuration Fundamentals Configuration Guide* for your release and the release-independent Cisco IOS Configuration Fundamentals Command Reference

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Feature Information for Using AutoInstall to Remotely Configure a Cisco Networking Device

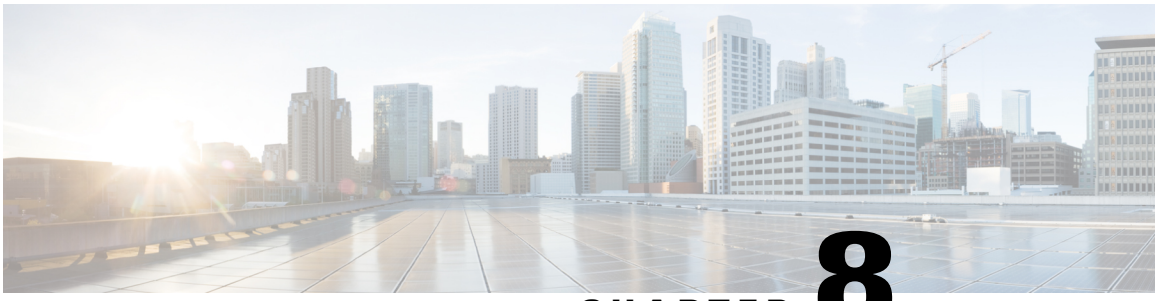
The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 12: Feature Information for Using AutoInstall to Remotely Set Up a Cisco Netowrking Device

Feature Name	Releases	Feature Configuration Information
AutoInstall Using DHCP for LAN Interfaces	Cisco IOS XE Release 2.1	The AutoInstall Using DHCP for LAN Interfaces feature enhances the benefits of AutoInstall by replacing the use of the Bootstrap Protocol (BOOTP) with the use of the Dynamic Host Configuration Protocol (DHCP) for Cisco IOS AutoInstall over LAN interfaces (specifically Fast Ethernet, Token Ring, and FDDI interfaces). In Cisco IOS XE Release 2.1, this feature was introduced on Cisco ASR 1000 Series Routers.

Feature Name	Releases	Feature Configuration Information
AutoInstall Support for TCL Script	Cisco IOS XE Release 3.3SE	The AutoInstall Using TCL Script feature enhances the AutoInstall feature by providing more flexibility in the installation process. This feature allows the users to program the device to get information about what to download, and to choose the type of file server, and the required file transfer protocol



CHAPTER 8

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the see the Feature Information Table at the end of this document. .

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn . An account on Cisco.com is not required.

Prerequisites for AutoInstall Using DHCP for LAN Interfaces

- You have read Overview: Basic Configuration of a Cisco Networking Device module in the *Cisco IOS Configuration Fundamentals Configuration Guide*.
- This document is written specifically for networking devices running Cisco IOS Release 12.4(1) or newer. However most of the information in this document can be used to configure networking devices that support AutoInstall and are not running Cisco IOS release 12.4(1) or newer. The two key differences that you must allow for are:
    - Some Cisco networking devices use a DHCP client identifier format that is different from the format used by networking devices running Cisco IOS release 12.4(1) or newer. This document only explains the DHCP client identifier format used by networking devices running Cisco IOS release 12.4(1) or newer. Use the process described in “Determining the Value for the DHCP Client

Identifier Automatically: Example" section on page 36 to determine the DHCP client identifier format that your Cisco networking device is using.

- No configuration file resides in NVRAM on the networking device that is being configured with AutoInstall.
- The configuration files that you want to load on to the networking device using AutoInstall reside on a TFTP server that is connected to the network. In most cases there is more than one file; for example, a network file with the IP-to-hostname mappings and a device-specific configuration file.
- You have someone at the remote site to connect the networking device that is being configured with AutoInstall to the network and power it on.
- The network has the IP connectivity necessary to permit the networking device to load configuration files from the TFTP server during the AutoInstall process.
- A DHCP server is available on the network to provide IP addresses to networking devices that are using AutoInstall over a LAN connection.

Restrictions for AutoInstall Using DHCP for LAN Interfaces

- (Serial interfaces only) AutoInstall over a serial interface using either HDLC or Frame Relay can be performed only over the first serial port on a new device (serial interface 0 or serial interface x/0).
- (LAN interfaces only) Only LAN Token Ring interfaces that set ring speed with physical jumpers support AutoInstall.
- AutoInstall does not automatically run on a T1 interface. For AutoInstall to work on a T1 interface, you have to manually configure the T1 interface to create a serial interface and then assign an IP address and network mask to that serial interface.

Information About Autoinstall Using DHCP for LAN Interfaces

AutoInstall Overview

AutoInstall can be used to load a final full configuration, or a partial temporary configuration, on to a networking device that is being configured with AutoInstall.

Tip When you use AutoInstall to load a partial temporary configuration, you must finish configuring the device manually.

Services and Servers Used by AutoInstall Dynamic Assignment of IP Addresses

The network must be able to provide the dynamic assignment of an IP address to the networking device that is being configured with AutoInstall. The type of IP address assignment server that is used depends on the type of connection that the networking that is being configured with AutoInstall has to the network.

AutoInstall uses these types of IP address servers:

DHCP Servers

Networking devices using AutoInstall over a LAN connection require a DHCP server to provide an IP address dynamically. This requirement applies to Ethernet, Token Ring, and FDDI interfaces. The network must be configured to provide IP connectivity between the DHCP server and any devices that are using AutoInstall over LAN connections.

DHCP (defined in RFC 2131) is an extension of the functionality provided by the Bootstrap Protocol (defined in RFC 951). DHCP provides the framework for passing configuration information to hosts on a TCP/IP network. DHCP adds the capability of automatic allocation of reusable network addresses and additional configuration options such as a router (gateway) IP address, a TFTP server IP address, the name of a boot file to load, and the domain name to use. DHCP servers can be configured on routers, UNIX servers, Microsoft Windows-based servers, and other platforms.

DHCP servers typically assign IP addresses from a pool of IP addresses randomly. It is possible for a device that uses DHCP to obtain its IP address to have a different IP address every time it is connected to the network. This behavior creates a problem for the AutoInstall process when you want to ensure that a particular device is assigned a specific hostname during the AutoInstall process. For example, if you are installing routers on different floors in a remote site and each router is supposed to be assigned a name that indicates its location, such as **ChicagoHQ-1st** and **ChicagoHQ-2nd**, you need to ensure that each device gets the IP address that will be mapped to its correct hostname.

The process of ensuring that a device is assigned a specific IP address is referred to as *creating a reservation*. A reservation is a manually configured relationship between an IP address and a physical layer address of a LAN interface on the device. Many Cisco IOS-based devices do not use their MAC address when they request an IP address via DHCP. They use a much longer client identifier instead. Due to the complexity of identifying the client identifier so that you can preconfigure a reservation, and the complexity of finding out if the new device uses its MAC address or the client identifier, we recommend that you allow a new device to obtain an IP address without using a DHCP reservation first in order to discover if the device is using its MAC address or a client identifier. When you have learned how the new device is identifying itself to the DHCP server, you can make a note of the format and create a reservation for it. The next time the new device is rebooted it should obtain the IP address that you reserved to ensure that the new device is assigned the correct hostname. Refer to the information on creating DHCP reservations that was provided with your DHCP server software. The process for creating reservations using Cisco IOS based DHCP servers is explained in the Using AutoInstall to Set Up Devices Connected to LANs section. This section includes instructions for identifying the client identifier before the device is connected to the network so that you can preconfigure the DHCP reservations.

Note This document uses a Cisco router as the DHCP server for using AutoInstall to configure LAN-connected networking devices. If you are using a different device as your DHCP server ensure that you have the user documentation for it available in the event that you need help configuring it.

Note There are several configuration parameters such as TFTP server addresses, DNS server addresses, domain names and so on, that can be provided to LAN-connected clients by DHCP servers during the process of assigning IP addresses to clients. These parameters are not required by AutoInstall, therefore they are not included in this document. If you know how to use these parameters, you can include them in your DHCP server configuration when you are using AutoInstall to set up your networking devices.

For more information on DHCP services visit the IETF RFC site (http://www.ietf.org/rfc.html) and look for RFCs about DHCP. Most server operating systems support DHCP servers. Refer to the documentation that was provided with your operating system for more information.

Services and Servers Used by AutoInstall IP-to-Hostname Mapping

If you want the networking device to load a full configuration file during the AutoInstall process, the networking device must be able to determine its hostname so that it can request the configuration file that you created specifically for it.

The following caveats apply to the provisioning of IP address to hostname mapping for AutoInstall:

- Any networking device that is being configured with AutoInstall can determine its hostname by loading one of the AutoInstall network configuration files (network-confg or cisconet.cfg) from the TFTP server that contain the **iphost***hostnameip-address* commands. For example, to map host R3 to IP address 198.162.100.3, the network-confg or cisconet.cfg file must contain the **iphostr3198.162.100.3** command.
- A networking device that is being configured with AutoInstall over a LAN interface can also determine its hostname by querying a DNS server. If the DNS server is not connected to the same LAN the device must learn the IP address of the DNS server from the DHCP server during the process of obtaining its dynamically assigned IP address from the DHCP server.

DNS Servers

DNS servers are used to provide a network service that maps hostnames to IP addresses and IP addresses to hostnames (reverse DNS lookups). Anytime that you use a hostname to initiate an IP connection to a host, your PC must determine the IP address that is assigned to the hostname that you want to contact. For example, when you visit Cisco's website (http://www.cisco.com/) your PC sends a DNS query to a DNS server to discover the current IP address that can be used to contact Cisco's website.

For more information on DNS services visit the IETF RFC site (http://www.ietf.org/rfc.html) and look for RFCs about DNS. The Name Server LookUp tool (nslookup) is very useful for learning more about DNS. There are several excellent websites available about nslookup that you can find by searching for them.

Services and Servers Used by AutoInstall Storage and Transmission of Configuration Files

TFTP is a protocol used to transfer files between devices on a network. A TFTP server is a device that uses TFTP to transfer files to devices. TFTP servers can be configured on UNIX servers, Microsoft Windows-based PCs and servers, and other platforms.

Tip If you do not have a TFTP server available you can configure a Cisco IOS-based router as a TFTP server using the **tftp-serverfile-system**:*filename* command. Refer to the Configuring Basic File Transfer Services feature for more information on configuring your router as a TFTP server.

Cisco routers use TFTP to load the configuration files that are required for AutoInstall. You must have a TFTP server deployed in your network to provide file storage and file transmission services to the devices that will be using AutoInstall.

For more information on TFTP services visit the IETF RFC site (http://www.ietf.org/rfc.html) and look for RFCs about TFTP. There are several excellent websites available about TFTP that you can find by searching for them. Several freeware and shareware versions of TFTP servers for various operating systems and hardware platforms are available from the Internet.

The following caveats apply to the provisioning of TFTP servers for AutoInstall:

- Devices using AutoInstall over a LAN--If the TFTP server and the devices using AutoInstall are on different LAN segments, you must either configure the **iphelper-address** *address* command on all of the interfaces that will receive TFTP session initialization requests from the devices that are using AutoInstall.
- Devices using AutoInstall over a WAN--If the devices using AutoInstall are connected to a WAN, you must configure the **iphelper-address** *address* command on all of the interfaces that will receive TFTP session initialization requests from devices that are using AutoInstall.

ip helper-address

If the new device does not learn the IP address of the TFTP server via DHCP option 150, it will transmit the TFTP session initialization requests as network layer broadcasts using the IP destination broadcast address of 255.255.255.255. Routers block network layer broadcast datagrams which prevents the TFTP session initialization requests from reaching the TFTP server, and AutoInstall will fail. The solution to this problem is to use the **ip helper-address** *address* command. The **ip helper-address** *address* command changes the broadcast address of TFTP session initialization request from 255.255.255.255 to the address that is configured with the *address* argument. For example, the **ip helper-address 172.16.29.252** command will change IP destination broadcast address of 255.255.255.255 to 172.16.29.252.

Networking Devices Used by AutoInstall

Device That Is Being Configured with AutoInstall

A device that is being configured with AutoInstall can be any Cisco IOS-based router that supports AutoInstall and does not have a configuration file in its NVRAM.

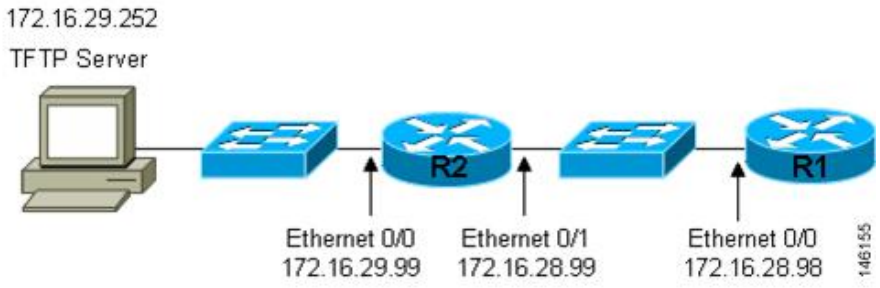
Staging Router DHCP/TFTP

A staging router acts as an intermediary between the TFTP server (to which it must have IP connectivity) and a device that is being configured with AutoInstall when the new device and the TFTP server are connected to different networks. In the figure below R1 requires a staging router because it is connected to a different LAN segment than the TFTP server.

Staging routers are required in the following situations:

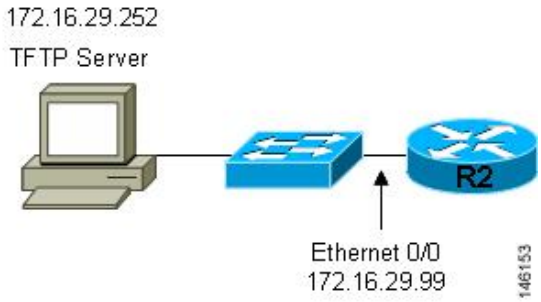
- Devices using AutoInstall over a LAN--If the TFTP and/or DHCP servers and the devices using AutoInstall are on different LAN segments you must use a staging router.

Figure 8: Example of AutoInstall That Requires a Staging Router



Staging routers are not required when the new device that is being configured with AutoInstall is connected to the same LAN segment as the TFTP and DHCP servers. In the figure below R2 does not require a staging server to use AutoInstall because it is on the same LAN segment as the TFTP server.

Figure 9: Example of AutoInstall That Does Not Require a Staging Router



Configuration Files Used by AutoInstall

A configuration file executes predefined commands and settings that enable a device to function in a network. The type of configuration file you choose determines many aspects of how you set up the network for AutoInstall.

These types of files are used by AutoInstall:

Network Configuration File

The network configuration file is the first file that the AutoInstall process attempts to use. After the device has obtained an IP address it will try to discover its hostname by attempting to download a network configuration file that contains IP address to host name mappings.

If you want the device to learn its hostname from the network-confg file so that it can download a host-specific configuration file, you must add an entry for the device in the network-confg network configuration file. The

syntax for the entry is **iphost***hostnameip-address* where *hostname* is the name that you want the host to use and *ip-address* is the address that the host will receive from the IP address server. For example, if you want the new device to use the name Australia, and the IP address that was dynamically assigned the new device is 172.16.29.103, you need to create an entry in the network configuration file that contains the **iphostaustralia172.16.29.103**command.

The file names used for the network configuration file are network-confg or cisconet.cfg. Routers running AutoInstall will try to load the network-confg from the TFTP server first. If the network-confg is not found on the TFTP server, the AutoInstall process will attempt to load the cisconet.cfg file. The cisconet.cfg filename was used by DOS-based TFTP servers that only supported the old 8.3 file naming convention. We recommend that you use the network-confg filename to avoid the delay that is created when AutoInstall has to timeout attempting to load the network-confg before it attempts to load the cisconet.cfg file.

If you use AutoInstall to set up multiple devices, you can create one network configuration file that contains an entry for each of the devices.

Host-Specific Configuration File

Host-specific configuration files are a full configuration for each new device. If you decide to use host-specific files, you must create a separate file for each new device that you are using AutoInstall to set up.

The filenames used for the host-specific configuration files are *name-confg* or *name.cfg* where the word name is replaced by the hostname of the router. For example, the filename for a router named hqrouter is hqrouter-confg or hqrouter.cfg.

Routers running AutoInstall will try to load the host-specific configuration filename using the format *name-confg* from the TFTP server first. If the *name-confg* file is not found on the TFTP server, the AutoInstall process will attempt to load the *name.cfg* file. The *name.cfg* file name format was used by DOS based TFTP servers that only supported the old 8.3 file naming convention. We recommend that you use the *name-confg* filename to avoid the delay that is created when AutoInstall has to timeout attempting to load the *name-confg* before it attempts to load the *name.cfg* file.

Tip If you use the *name.cfg* format for host-specific configuration files the filenames for hostnames that are longer than 8 characters must be truncated to the first eight characters. For example, the filename for a device with the hostname australia must be truncated to australi.cfg. When AutoInstall maps the IP address assigned to the new router to its hostname of australia in the network configuration file, AutoInstall will attempt to download a host-specific file with the name australi.cfg after it fails to load the host-specific filename austrailia-confg.

Tip Cisco recommends that you use the host-specific file option for setting up new devices to ensure that each new device is set up properly.

Default Configuration File (Optional)

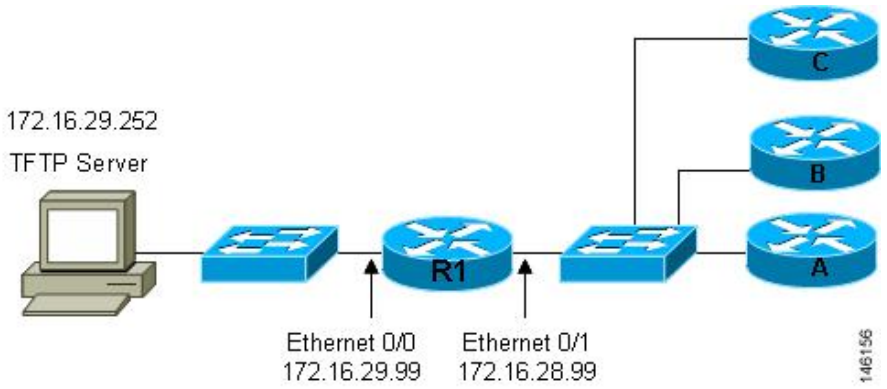
A default configuration file, which includes minimum configuration information allows you to telnet to the new device and configure it manually.

Tip If the new device has learned its hostname after it loaded the network configuration file the default configuration file is not used. You must use the host-specific file instead to configure features such as passwords for remote CLI sessions.

The figure below is an example of using the default configuration file to stage new routers for remote manual configuration. Routers A, B, and C are new routers that will be added to the network one at a time. You connect the first router and wait for it to load the default configuration file. The default configuration file must have enough information in it to allow the new router to communicate with the PC that you will be using to finish its configuration using a Telnet session. After the default configuration file is loaded on the new router, you can use Telnet to connect to the router to complete its configuration. You must assign a new, unique IP address to its interfaces so that the default configuration file can be used for configuring the next router.

Caution Failure to change the IP addresses in the router that you are configuring remotely with Telnet will result in duplicate IP addresses on the LAN when the next router loads the default configuration file. In this situation you will not be able to use Telnet to connect to either router. You must disconnect one of the routers before you can resolve this problem.

Figure 10: Example of Using the Default Configuration File to Stage Routers for Remote Manual Configuration



Tip You must include the commands for configuring passwords for remote Telnet access and access to privileged EXEC mode if you are going to access the routers remotely to complete their configurations save their configuration files to NVRAM.

The filenames used for the default network configuration file are router-confg or router.cfg. Routers running AutoInstall will try to load the router-confg from the TFTP server first. If the router-confg is not found on the TFTP server the AutoInstall process will attempt to load the router.cfg file. The router.cfg file name was used by DOS-based TFTP servers that only supported the old 8.3 file naming convention. We recommend that you use the router-confg filename to avoid the delay that is created when AutoInstall has to timeout while attempting to load the router-confg before it attempts to load the router.cfg file.

If you are using AutoInstall to configure LAN-attached devices, you can specify a different default boot filename in DHCP Option 067.

Configuration Options for Autoinstall using DHCP

You can provision your network to support AutoInstall using several different combinations of devices and services. For example:

- You can provision all of the services required for AutoInstall on one network server, or you can provision each service on a different network server.
- You can provision the DHCP service on a Cisco router.
- The device using AutoInstall can determine its IP address from a DNS server, or you can use one of the AutoInstall network configuration files (network-confg or cisconet.cfg) that contain the **ip host** *hostname ip-address* commands.
- You can use provision AutoInstall to load a full configuration or a partial configuration onto a device that is using AutoInstall.

This module focuses on some of the most common methods for provisioning AutoInstall. Refer to the How to Use AutoInstall to Remotely Configure Cisco Networking Devices module for information on the most common methods for provisioning AutoInstall.

Benefits of Using AutoInstall to Remotely Configure a Cisco Networking Device

AutoInstall facilitates the deployment of Cisco routers by allowing you to manage the setup procedure for routers from a central location. The person responsible for physically installing the router does not require specific networking skills. The ability to physically install the router, connect the power and networking cables, and power it on are the only skills required by the installer. The configuration files are stored and managed on a central TFTP server. By using AutoInstall one skilled network technician based at a central site can manage the deployment of several routers in a short period of time.

AutoInstall Using DHCP for LAN Interfaces

The AutoInstall Using DHCP for LAN Interfaces feature enhances the benefits of AutoInstall by replacing the use of the Bootstrap Protocol (BOOTP) with the use of the Dynamic Host Configuration Protocol (DHCP) for Cisco IOS AutoInstall over LAN interfaces (specifically Ethernet, Token Ring, and FDDI interfaces).

DHCP (defined in RFC 2131) is an extension of the functionality provided by the BOOTP (defined in RFC 951). DHCP provides the framework for passing configuration information to hosts on a TCP/IP network. DHCP adds the capability of automatic allocation of reusable network addresses and additional configuration options. In Cisco IOS Release 12.1(5)T, and later releases, the IP address procurement phase of the AutoInstall process is now accomplished using DHCP for Ethernet, Token Ring, and FDDI interfaces. Prior to this release, IP addresses for LAN interfaces were obtained using BOOTP or RARP during the AutoInstall process. Additionally, this feature allows for the uploading of configuration files using unicast TFTP.

Using AutoInstall to Remotely Configure Cisco Networking Devices

AutoInstall enables remote, automatic configuration of networking devices. AutoInstall is typically used to set up new networking devices remotely. You can, however, use AutoInstall to configure existing networking devices after you remove the configuration file from their NVRAM. The AutoInstall process uses preexisting configuration files that are stored on a TFTP server.

In this module the term networking device means a router that runs Cisco IOS software. Also, the following terms are used interchangeably:

- initial configuration and startup configuration
- *set up* and *configure*

Determining the Value for the DHCP Client Identifier Manually

If you want to determine the value for the client identifiers automatically, you do not need to perform this task. Proceed to the Determining the Value for the DHCP Client Identifier Automatically module.

You must know the MAC address of the Fast Ethernet interface that will be used to connect the router to the LAN during the AutoInstall process to determine the client identifier manually. This requires connecting a terminal to the router, and powering it on, so that you can enter the **show interface** *interface-type interface-number* command.

The client-identifier looks like this:

0063.6973.636f.2d30.3030.362e.3533.6237.2e38.6537.312d.4661.332f.30

The format is *nullcisco-0006.53b7.8e71-fa3/0* where *0006.53b7.8e71* is the MAC address and *fa3/0* is the short interface name for the interface that the IP address request is made for.

The values for the short-if-name field can be obtained from an SNMP workstation with the Cisco MIBs installed. This is an example of how to map ifIndex to an interface on Cisco IOS:

```
snmpwalk -c public ponch ifName
IF-MIB::ifName.1 = STRING: AT2/0
IF-MIB::ifName.2 = STRING: Et0/0
IF-MIB::ifName.3 = STRING: Se0/0
IF-MIB::ifName.4 = STRING: BR0/0
```

Use the **show interface** *interface-type interface-number* command to display the information and statistics for a Fast Ethernet interface.

```
R6> show interface fastethernet 3/0
FastEthernet3/0 is up, line protocol is up
  Hardware is AmdFE, address is 0006.53b7.8e71 (bia 0006.53b7.8e71)
.
.
.
R6>
```

The MAC address for FastEthernet 3/0 on R6 is 0006.53b7.8e71. The format of the client identifier for this interface is nullcisco-0006.53b7.8e71-fa3/0.

Note The short interface name for Fast Ethernet interfaces is fa.

The table below shows the values for converting characters to their hexadecimal equivalents. The last row in the second table below shows the client identifier for Fast Ethernet 3/0 on R6 (nullcisco-0006.53b7.8e71-fa3/0).

Table 13: Hexadecimal to Character Conversion Chart

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
00	NUL	1a	SUB	34	4	4e	N	68	h
01	SOH	1b	ESC	35	5	4f	O	69	I
02	STX	1c	FS	36	6	50	P	6a	j
03	ETX	1d	GS	37	7	51	Q	6b	k
04	EOT	1e	RS	38	8	52	R	6c	l
05	ENQ	1f	US	39	9	53	S	6d	m
06	ACK	20		3a	:	54	T	6e	n
07	BEL	21	!	3b	;	55	U	6f	o
08	BS	22	"	3c	<	56	V	70	p
09	TAB	23	#	3d	=	57	W	71	q
0A	LF	24	$	3e	>	58	X	72	r
0B	VT	25	%	3f	?	59	Y	73	s
0C	FF	26	&	40	@	5a	Z	74	t
0D	CR	27	‘	41	A	5b	[	75	u
0E	SO	28	(	42	B	5c	\	76	v
0F	SI	29	)	43	C	5d	]	77	w
10	DLE	2a	*	44	D	5e	^	78	x
11	DC1	2b	+	45	E	5f	_	79	y
12	DC2	2c	,	46	F	60	`	7a	z
13	DC3	2d	-	47	G	61	a	7b	{

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
14	DC4	2e	.	48	H	62	b	7c	|
15	NAK	2f	/	49	I	63	c	7D	}
16	SYN	30	0	4a	J	64	d	7e	~
17	ETB	31	1	4b	K	65	e	7f	D
18	CAN	32	2	4c	L	66	f		
19	EM	33	3	4d	M	67	g		

Table 14: Conversion of nullcisco-0006.53b7.8e71-fa3/0 To A Client Identifier

00	c	i	s	c	o	-	0	0	0	6	.	5	3	b	7	.	8	e	7	1	-	f	a	3	/	0
00	63	69	73	63	6f	2d	30	30	30	36	2e	35	33	62	37	2e	38	65	37	31	2d	46	61	33	2f	30

R4

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R4.

```
R4> show interface FastEthernet 0/0
FastEthernet0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb0e (bia 00e0.1eb8.eb0e)
```

The MAC address for Fast Ethernet 0/0 on R4 is 00e0.1eb8.eb0e. The format of the client identifier for this interface is nullcisco-00e0.1eb8.eb0e-et0.

Note The short interface name for Fast Ethernet interfaces is et.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R4 is shown in the last row of the table below.

Table 15: Conversion of null.cisco-00e0.1eb8.eb0e-et0 To A Client Identifier for R4

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	0	e	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	30	65	2d	45	74	30

R3

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R3.

```
R3> show interface FastEthernet 0/0
FastEthernet0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb73 (bia 00e0.1eb8.eb73)
```

The MAC address for Fast Ethernet 0/0 on R3 is 00e0.1eb8.eb73. The format of the client identifier for this interface is: nullcisco-00e0.1eb8.eb73-et0.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R3 is shown in the last row of the table below.

Table 16: Conversion of null.cisco-00e0.1eb8.eb73-et0 To A Client Identifier for R3

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	7	3	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	37	33	2d	45	74	30

R2

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R2.

```
R2> show interface Fast Ethernet 0/0
FastEthernet0/0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb09 (bia 00e0.1eb8.eb09)
```

The MAC address for Fast Ethernet 0/0 on R2 is 00e0.1eb8.eb09. The format of the client identifier for this interface is nullcisco-00e0.1eb8.eb09-et0.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R2 is shown in the last row of the table below

Table 17: Conversion of null.cisco-00e0.1eb8.eb09-et0 To A Client Identifier for R2

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	0	9	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	30	39	2d	45	74	30

You have now determined the values for the client identifiers on each router. The final step is to add a period after each group of four characters working from the left to the right as shown below:

- R4-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30
- R3-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30
- R2-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30

What to Do Next

Refer to the "Managing Connections, Menus, and System Banners" chapter for more information on ending sessions and closing connections.

Determining the Value for the DHCP Client Identifier Automatically

If you determined the value for the client identifiers manually, you do not need to perform this task. Proceed to the Creating a Private DHCP Pool for Each of the Routers Example section.

This task will create a DHCP server on R1 that will provide only one IP address. This IP address will used by each new router in sequence while you determine the value of the router's client identifier. By limiting the IP address scope to a single IP address you avoid any possible confusion about which router you are working on. If somebody powers up another router that attempts to start the AutoInstall process, it will not be able to obtain an IP address.

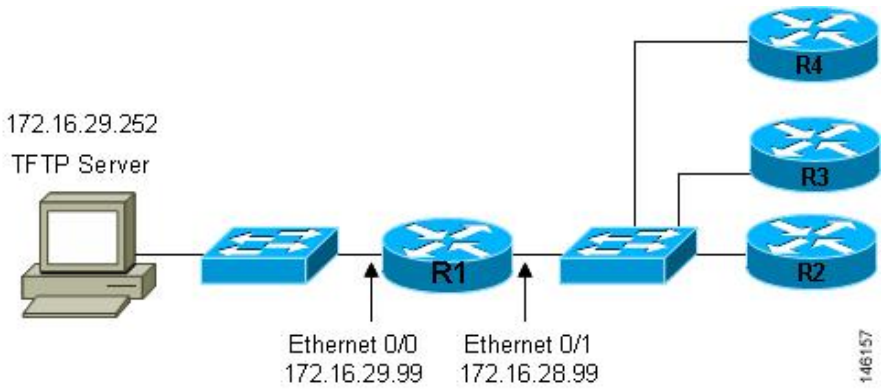
Tip Do not place the network-confg or router configuration files (r4-confg, r3-confg, or r2-confg) in the root directory of the TFTP server yet. You do not want any of the routers to load these files until you have ensured that each router will obtain the correct IP address from the DHCP server so that the router will load the correct configuration file.

This task is broken down into subtasks. See the Determining the Value for the DHCP Client Identifier Manually section for more information.

Using AutoInstall to Set Up Devices Connected to LANs Example

This task uses the network in the figure below. This task will show how to use AutoInstall to setup routers R2, R3, and R4. Router R1 is the DHCP server that will be used to assign the IP address for Fast Ethernet 0/0 on the new routers during the AutoInstall process.

Figure 11: Network Topology for Assigning AutoInstall Configuration Files For Specific Devices



Every DHCP client has a unique DHCP client identifier. The DHCP client identifier is used by DHCP servers to keep track of IP address leases and for configuring IP address reservations. You need to know the DHCP

client identifier for each of the networking devices that you want to configure with AutoInstall so that you can configure the DHCP IP address reservations which will ensure that each device is provided with the correct IP address, and subsequently its unique configuration file. You can determine the DHCP client identifier manually or automatically.

To use AutoInstall to setup routers R2, R3, and R4, perform following tasks:

Determining the Value for the DHCP Client Identifier Manually

If you want to determine the value for the client identifiers automatically, you do not need to perform this task. Proceed to the Determining the Value for the DHCP Client Identifier Automatically module.

You must know the MAC address of the Fast Ethernet interface that will be used to connect the router to the LAN during the AutoInstall process to determine the client identifier manually. This requires connecting a terminal to the router, and powering it on, so that you can enter the **show interface** *interface-type interface-number* command.

The client-identifier looks like this:

0063.6973.636f.2d30.3030.362e.3533.6237.2e38.6537.312d.4661.332f.30

The format is *nullcisco-0006.53b7.8e71-fa3/0* where *0006.53b7.8e71* is the MAC address and *fa3/0* is the short interface name for the interface that the IP address request is made for.

The values for the short-if-name field can be obtained from an SNMP workstation with the Cisco MIBs installed. This is an example of how to map ifIndex to an interface on Cisco IOS:

```
snmpwalk -c public ponch ifName
IF-MIB::ifName.1 = STRING: AT2/0
IF-MIB::ifName.2 = STRING: Et0/0
IF-MIB::ifName.3 = STRING: Se0/0
IF-MIB::ifName.4 = STRING: BR0/0
```

Use the **show interface** *interface-type interface-number* command to display the information and statistics for a Fast Ethernet interface.

```
R6> show interface fastethernet 3/0
FastEthernet3/0 is up, line protocol is up
  Hardware is AmdFE, address is 0006.53b7.8e71 (bia 0006.53b7.8e71)
.
.
.
R6>
```

The MAC address for FastEthernet 3/0 on R6 is 0006.53b7.8e71. The format of the client identifier for this interface is nullcisco-0006.53b7.8e71-fa3/0.

Note The short interface name for Fast Ethernet interfaces is fa.

The table below shows the values for converting characters to their hexadecimal equivalents. The last row in the second table below shows the client identifier for Fast Ethernet 3/0 on R6 (nullcisco-0006.53b7.8e71-fa3/0).

Table 18: Hexadecimal to Character Conversion Chart

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
00	NUL	1a	SUB	34	4	4e	N	68	h

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
01	SOH	1b	ESC	35	5	4f	O	69	I
02	STX	1c	FS	36	6	50	P	6a	j
03	ETX	1d	GS	37	7	51	Q	6b	k
04	EOT	1e	RS	38	8	52	R	6c	l
05	ENQ	1f	US	39	9	53	S	6d	m
06	ACK	20		3a	:	54	T	6e	n
07	BEL	21	!	3b	;	55	U	6f	o
08	BS	22	"	3c	<	56	V	70	p
09	TAB	23	#	3d	=	57	W	71	q
0A	LF	24	$	3e	>	58	X	72	r
0B	VT	25	%	3f	?	59	Y	73	s
0C	FF	26	&	40	@	5a	Z	74	t
0D	CR	27	‘	41	A	5b	[	75	u
0E	SO	28	(	42	B	5c	\	76	v
0F	SI	29	)	43	C	5d	]	77	w
10	DLE	2a	*	44	D	5e	^	78	x
11	DC1	2b	+	45	E	5f	_	79	y
12	DC2	2c	,	46	F	60	`	7a	z
13	DC3	2d	-	47	G	61	a	7b	{
14	DC4	2e	.	48	H	62	b	7c	|
15	NAK	2f	/	49	I	63	c	7D	}
16	SYN	30	0	4a	J	64	d	7e	~
17	ETB	31	1	4b	K	65	e	7f	D
18	CAN	32	2	4c	L	66	f		

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
19	EM	33	3	4d	M	67	g		

Table 19: Conversion of nullcisco-0006.53b7.8e71-fa3/0 To A Client Identifier

00	c	i	s	c	o	-	0	0	0	6	.	5	3	b	7	.	8	e	7	1	-	f	a	3	/	0
00	63	69	73	63	6f	2d	30	30	30	36	2e	35	33	62	37	2e	38	65	37	31	2d	46	61	33	2f	30

R4

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R4.

```
R4> show interface FastEthernet 0/0
FastEthernet0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb0e (bia 00e0.1eb8.eb0e)
```

The MAC address for Fast Ethernet 0/0 on R4 is 00e0.1eb8.eb0e. The format of the client identifier for this interface is nullcisco-00e0.1eb8.eb0e-et0.

Note The short interface name for Fast Ethernet interfaces is et.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R4 is shown in the last row of the table below.

Table 20: Conversion of null.cisco-00e0.1eb8.eb0e-et0 To A Client Identifier for R4

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	0	e	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	30	65	2d	45	74	30

R3

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R3.

```
R3> show interface FastEthernet 0/0
FastEthernet0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb73 (bia 00e0.1eb8.eb73)
```

The MAC address for Fast Ethernet 0/0 on R3 is 00e0.1eb8.eb73. The format of the client identifier for this interface is: nullcisco-00e0.1eb8.eb73-et0.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R3 is shown in the last row of the table below.

Table 21: Conversion of null.cisco-00e0.1eb8.eb73-et0 To A Client Identifier for R3

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	7	3	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	37	33	2d	45	74	30

R2

Use the **show interface** *interface-type interface-number* command to display the information and statistics for Fast Ethernet 0/0 on R2.

```
R2> show interface Fast Ethernet 0/0
FastEthernet0/0 is up, line protocol is up
  Hardware is Lance, address is 00e0.1eb8.eb09 (bia 00e0.1eb8.eb09)
```

The MAC address for Fast Ethernet 0/0 on R2 is 00e0.1eb8.eb09. The format of the client identifier for this interface is nullcisco-00e0.1eb8.eb09-et0.

Using the values for converting characters to their hexadecimal equivalents in the first table above, the client identifier for Fast Ethernet 0/0 on R2 is shown in the last row of the table below

Table 22: Conversion of null.cisco-00e0.1eb8.eb09-et0 To A Client Identifier for R2

00	c	i	s	c	o	-	0	0	e	0	.	1	e	b	8	.	e	b	0	9	-	e	t	0
00	63	69	73	63	6f	2d	30	30	65	30	2e	31	65	62	38	2e	65	62	30	39	2d	45	74	30

You have now determined the values for the client identifiers on each router. The final step is to add a period after each group of four characters working from the left to the right as shown below:

- R4-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30
- R3-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30
- R2-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30

Determining the Value for the DHCP Client Identifier Automatically

If you determined the value for the client identifiers manually, you do not need to perform this task. Proceed to the Creating a Private DHCP Pool for Each of The Routers module.

This task will create a DHCP server on R1 that will provide only one IP address. This IP address will used by each new router in sequence while you determine the value of the router's client identifier. By limiting the IP address scope to a single IP address you avoid any possible confusion about which router you are working on. If somebody powers up another router that attempts to start the AutoInstall process, it will not be able to obtain an IP address.

Tip Do not place the network-confg or router configuration files (r4-confg, r3-confg, or r2-confg) in the root directory of the TFTP server yet. You do not want any of the routers to load these files until you have ensured that each router will obtain the correct IP address from the DHCP server so that the router will load the correct configuration file.

This task is broken down into sub-tasks to make it easier to follow (all sub-tasks are required):

Configuring IP on the Interfaces on R1

Configure IP addresses on the Fast Ethernet interfaces. Configure the **ip helper-address** *ip-address* command on Fast Ethernet 0/1.

```
!
interface FastEthernet0/0
 ip address 172.16.29.99 255.255.255.0
!
interface FastEthernet0/1
 ip address 172.16.28.99 255.255.255.0
 ip helper-address 172.16.29.252
!
```

Configuring a DHCP Pool on R1

Configure these commands to setup the temporary DHCP server on R1.

Note This should be the only DHCP server in operation on R1. This should be the only DHCP server that is accessible by the routers that you will be using AutoInstall to setup.

```
!
ip dhcp pool get-client-id
   network 172.16.28.0 255.255.255.0
!
```

Excluding All But One of the IP Addresses from the DHCP Pool on R1

You need to ensure that there is only one IP address available from the DHCP server at any time. Configure the following command to exclude every IP address except 172.16.28.1 from the DHCP pool.

```
!
ip dhcp excluded-address 172.16.28.2 172.16.28.255
!
```

Verifying The Configuration on R1

Verify that the configuration file for R1 has a DHCP server pool configured to provide a single IP address (172.16.28.1) to a DHCP client.

Verify that the configuration file has the IP addresses for the Fast Ethernet interfaces and the **ip helper-address** *ip-address* command.

```
!
ip dhcp excluded-address 172.16.28.2 172.16.28.255
!
ip dhcp pool get-client-id
   network 172.16.28.0 255.255.255.0
!
interface FastEthernet0/0
 ip address 172.16.29.99 255.255.255.0
!
interface FastEthernet0/1
 ip address 172.16.28.99 255.255.255.0
 ip helper-address 172.16.29.252
!
```

Enabling debug ip dhcp server events on R1

You use the display output from the **debug ip dhcp server events** command on the terminal connected to R1 to identify the value of the client identifier for each router.

Enable the **debug ip dhcp server events** command on R1.

```
R1# debug ip dhcp server events
```

Identifying the Value for the Client Identifier on Each of the Routers

This step is repeated for each of the routers. You should only have one of the routers powered-on at any time. When you have identified the value of the client identifier field for the router, you will turn the router off and proceed to the next router.

R4

Connect R4 to the Fast Ethernet network and power it on. The following message will be displayed on the terminal connected to R1 when R4 is assigned the IP address 172.16.28.1.

```
DHCPD: assigned IP address 172.16.28.1 to client
0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30.
```

Copy the client identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30 to a text file and save it. Keep the text file open for the next two routers.

Turn off R4

Release the IP address binding for R4 from the DHCP pool on R1 using the **clear ip dhcp binding *** command on R1.

```
R1# clear ip dhcp binding *
R1#
01:16:11: DHCPD: returned 172.16.28.1 to address pool get-client-id.
```

R3

Connect R3 to the Fast Ethernet network and power it on. The following message will be displayed on the terminal connected to R1 when R3 is assigned the IP address 172.16.28.1.

```
DHCPD: assigned IP address 172.16.28.1 to client
0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30.
```

Copy the client identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30 to the text file and save it. Keep the text file open for the final router.

Turn off R3.

Release the IP address binding for R3 from the DHCP pool on R1 using the **clear ip dhcp binding *** command on R1.

```
R1# clear ip dhcp binding *
R1#
01:16:11: DHCPD: returned 172.16.28.1 to address pool get-client-id.
```

R2

Connect R2 to the Fast Ethernet network and power it on. The following message will be displayed on the terminal connected to R1 when R2 is assigned the IP address 172.16.28.1.

```
DHCPD: assigned IP address 172.16.28.1 to client
0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30.
```

Copy the client identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30 to the text file and save it.

Turn off R2

Release the IP address binding for R2 from the DHCP pool on R1 using the **clear ip dhcp binding *** command on R1.

```
R1# clear ip dhcp binding *
R1#
01:16:11: DHCPD: returned 172.16.28.1 to address pool get-client-id.
```

Client Identifiers for R4, R3, and R2

You have determined the values for the client identifiers on each router.

- R4-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30
- R3-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30
- R2-0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30

Removing the DHCP Pool on R1 for Network 172.16.28.0 24

The temporary DHCP pool on the router is no longer required, and must be removed.

```
R1(config)# no ip dhcp pool get-client-id
```

Removing the Excluded Address Range From R1

The command for excluding all of the IP addresses except 172.16.28.1 from the DHCP pool on the router is no longer required, and must be removed.

R1(config)# **no ip dhcp excluded-address 172.16.28.2 172.16.28.255**

Creating a Private DHCP Pool for Each of The Routers

You need to create the private DHCP address pools for each router to ensure that each router is assigned the IP address that maps to its host name in the network-conf file.

```
!
ip dhcp pool r4
   host 172.16.28.100 255.255.255.0
   client-identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.652d.4574.30
!
ip dhcp pool r3
   host 172.16.28.101 255.255.255.0
   client-identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6237.332d.4574.30
!
ip dhcp pool r2
   host 172.16.28.102 255.255.255.0
   client-identifier 0063.6973.636f.2d30.3065.302e.3165.6238.2e65.6230.392d.4574.30
```

Creating Configuration Files for Each Router

Create the configuration files for each router and place them in the root directory of the TFTP server.

Tip You must include the commands for configuring passwords for remote Telnet access and access to privileged EXEC mode if you are going to access the routers remotely to save their configuration files to NVRAM.

r2-confg

```
!
hostname R2
!
enable secret 7gD2A0
!
interface FastEthernet0/0
 ip address 172.16.28.102 255.255.255.0
!
interface Serial0/0
 ip address 192.168.100.1 255.255.255.252
 no shutdown
!
interface Serial0/1
 ip address 192.168.100.5 255.255.255.252
 no shutdown
!
no ip http server
ip classless
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0
!
line vty 0 4
 password 5Rf1k9
 login
!
end
```

r3-confg

```
!
hostname R3
!
enable secret 7gD2A0
!
interface FastEthernet0/0
 ip address 172.16.28.101 255.255.255.0
!
interface Serial0/0
 ip address 192.168.100.9 255.255.255.252
 no shutdown
!
interface Serial0/1
 ip address 192.168.100.13 255.255.255.252
 no shutdown
!
no ip http server
ip classless
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 FastEthernet0
!
line vty 0 4
 password 5Rf1k9
 login
!
end
```

r4-confg

```
!
hostname R3
!
enable secret 7gD2A0
!
interface FastEthernet0/0
 ip address 172.16.28.101 255.255.255.0
!
interface Serial0/0
 ip address 192.168.100.9 255.255.255.252
 no shutdown
!
interface Serial0/1
 ip address 192.168.100.13 255.255.255.252
 no shutdown
!
no ip http server
ip classless
ip default-network 0.0.0.0
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0
!
line vty 0 4
 password 5Rf1k9
 login
!
end
```

Creating the network-confg file

Create the network-confg file with the **ip host** *hostname ip-address* commands that map the IP addresses that you will be assigning with the DHCP server to the hostname.

```
ip host r4 172.16.28.100
ip host r3 172.16.28.101
ip host r2 172.16.28.102
```

Setting Up the Routers with AutoInstall

You are now ready to set up the three routers (R4, R3, and R2) using AutoInstall.

Connect a terminal to the routers if you want to monitor the progress of AutoInstall. Use Hyperterminal or a similar terminal emulation program on your PC, with the following terminal emulation settings, to connect to the device:

- 9600 baud
- 8 data bits, no parity, 1 stop bit
- No flow control

You should have the following files in the root directory of the TFTP server.

- network-confg
- r4-confg
- r3-confg
- r2-confg

The TFTP server must be running.

Power on each router.

Timesaver You can set up all three routers concurrently.

R4

The following is an excerpt of the messages that are displayed on R4's console terminal during the AutoInstall process:

```
Loading network-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 76 bytes]
Configuration mapped ip address 172.16.28.100 to r4
Loading r4-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 687 bytes]
```

R3

The following is an excerpt of the messages that are displayed on R3's console terminal during the AutoInstall process:

```
Loading network-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 76 bytes]
Configuration mapped ip address 172.16.28.101 to r3
Loading r3-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 687 bytes]
```

R2

The following is an excerpt of the messages that are displayed on R2's console terminal during the AutoInstall process:

```
Loading network-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 76 bytes]
Configuration mapped ip address 172.16.28.102 to r2
Loading r2-confg from 172.16.29.252 (via FastEthernet0/0): !
[OK - 687 bytes]
```

TFTP Server Log

The TFTP server log should contain messages similar to the following text.

```
Sent network-confg to (172.16.28.100), 76 bytes
Sent r4-confg to (172.16.28.100),687 bytes
Sent network-confg to (172.16.28.101), 76 bytes
Sent r3-confg to (172.16.28.101),687 bytes
Sent network-confg to (172.16.28.102), 76 bytes
Sent r2-confg to (172.16.28.102),687 bytes
```

Saving the Configuration Files on The Routers

You must save the running configurations on each router to the startup configuration to ensure that the routers retain their configurations if they are ever power cycled.

R4

```
R1# telnet 172.16.28.100
Trying 172.16.28.100 ... Open
User Access Verification
Password:
R4> enable
Password:
R4# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R4# exit
[Connection to 172.16.28.100 closed by foreign host]
R1#
```

R3

```
R1# telnet 172.16.28.101
Trying 172.16.28.101 ... Open
User Access Verification
Password:
R3> enable
Password:
R3# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R3# exit
[Connection to 172.16.28.101 closed by foreign host]
R1#
```

R2

```
R1# telnet 172.16.28.102
Trying 172.16.28.102 ... Open
User Access Verification
Password:
R2> enable
Password:
R2# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R2# exit
[Connection to 172.16.28.102 closed by foreign host]
R1#
```

Removing the Private DHCP Address Pools from R1

The final step in the AutoInstall process is to remove the private DHCP address pools from R1.

```
R1(config)# no ip dhcp pool r4
R1(config)# no ip dhcp pool r3
R1(config)# no ip dhcp pool r2
```

This is the final task, and step for Using AutoInstall to Setup Devices Connected to LANs.

Additional References

The following sections provide references related to using AutoInstall to remotely configure Cisco networking devices.

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Configuration Fundamentals commands	Cisco IOS Configuration Fundamentals Command Reference
Frame Relay-to-ATM Service Interworking (FRF.8)	• Frame Relay-ATM Interworking Supported Standards module in the *Cisco IOS Wide-Area Networking Configuration Guide* • Configuring Frame Relay-ATM Interworking module in the *Cisco IOS Wide-Area Networking Configuration Guide*
Overview of Cisco IOS setup mode and AutoInstall for configuring Cisco networking devices	Overview: Basic Configuration of a Cisco Networking Device module in the *Cisco IOS Configuration Fundamentals Configuration Guide*

Related Topic	Document Title
Using setup mode to configure a Cisco networking device	Using Setup Mode to Configure a Cisco Networking Device module in the *Cisco IOS Configuration Fundamentals Configuration Guide*

MIBs

MIB	MIBs Link
IF-MIB	The IFNAME object in the IF-MIB can be used to identify the values for the short interface names used in the DHCP Client Identifier for Cisco IOS devices when they are configured as DHCP clients. To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Standards and RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature	--

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

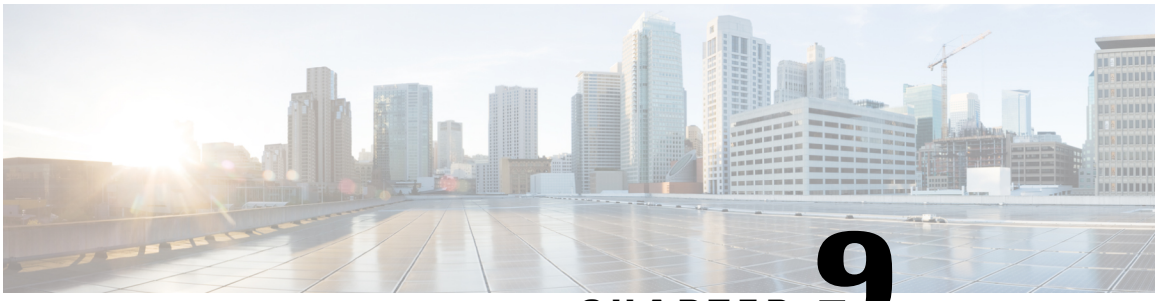
Feature Information for AutoInstall Using DHCP for LAN Interfaces

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 23: Feature Information for AutoInstall Using DHCP for LAN Interfaces

Feature Name	Releases	Feature Configuration Information
AutoInstall Using DHCP for LAN Interfaces	12.1(5)T 12.2(33)SRC	The AutoInstall Using DHCP for LAN Interfaces feature enhances the benefits of AutoInstall by replacing the use of the Bootstrap Protocol (BOOTP) with the use of the Dynamic Host Configuration Protocol (DHCP) for Cisco IOS AutoInstall over LAN interfaces (specifically Ethernet, Token Ring, and FDDI interfaces).



CHAPTER 9

Unique Device Identifier Retrieval

The Unique Device Identifier Retrieval feature provides the ability to retrieve and display the Unique Device Identifier (UDI) information from any Cisco product that has electronically stored such identity information.

History for Unique Device Identifier Retrieval Feature

Release	Modification
12.3(4)T	This feature was introduced.
12.0(27)S	This feature was integrated into Cisco IOS Release 12.0(27)S.
12.2(25)S	This feature was integrated into Cisco IOS Release 12.2(25)S.
12.2(27)SBC	This feature was integrated into Cisco IOS Release 12.2(27)SBC.
12.2(18)SXE5	This feature was integrated into Cisco IOS Release 12.2(18)SXE5.

Software images for Cisco 12000 series Internet routers have been deferred to Cisco IOS Release 12.0(27)S1.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn . You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

• Additional References, page 117

Prerequisites for Unique Device Identifier Retrieval

In order to use UDI retrieval, the Cisco product in use must be UDI-enabled. A UDI-enabled Cisco product supports five required Entity MIB objects. The five Entity MIB v2 (RFC-2737) objects are as follows:

- entPhysicalName
- entPhysicalDescr
- entPhysicalModelName
- entPhysicalHardwareRev
- entPhysicalSerialNum

Although the **showinventory** command may be available, using that command on devices that are not UDI-enabled will likely produce no output.

Information About Unique Device Identifier Retrieval

Unique Device Identifier Overview

Each identifiable product is an entity, as defined by the Entity MIB (RFC-2737) and its supporting documents. Some entities, such as a chassis, will have subentities like slots. An Ethernet switch might be a member of a superentity like a stack. Most Cisco entities that are orderable products will leave the factory with an assigned UDI. The UDI information is printed on a label that is affixed to the physical hardware device, and it is also stored electronically on the device in order to facilitate remote retrieval.

A UDI consists of the following elements:

- Product identifier (PID)
- Version identifier (VID)
- Serial number (SN)

The PID is the name by which the product can be ordered; it has been historically called the “Product Name” or “Part Number.” This is the identifier that one would use to order an exact replacement part.

The VID is the version of the product. Whenever a product has been revised, the VID will be incremented. The VID is incremented according to a rigorous process derived from Telcordia GR-209-CORE, an industry guideline that governs product change notices.

The SN is the vendor-unique serialization of the product. Each manufactured product will carry a unique serial number assigned at the factory, which cannot be changed in the field. This is the means by which to identify an individual, specific instance of a product.

Benefits of the Unique Device Identifier Retrieval Feature

- Identifies individual Cisco products in your networks.
- Reduces operating expenses for asset management through simple, cross-platform, consistent identification of Cisco products.
- Identifies PIDs for replaceable products.
- Facilitates discovery of products subject to recall or revision.
- Automates Cisco product inventory (capital and asset management).
- Provides a mechanism to determine the entitlement level of a Cisco product for repair and replacement service.

How to Retrieve the Unique Device Identifier

Retrieving the Unique Device Identifier

Perform this task to retrieve and display identification information for a Cisco product.

SUMMARY STEPS

1. **enable**
2. **show inventory** [**raw**] [*entity*

DETAILED STEPS

Step 1 **enable**
Enters privileged EXEC mode. Enter your password if prompted.

Example:

```
Router> enable
```

Step 2 **show inventory** [**raw**] [*entity*
Enter the**showinventory** command to retrieve and display information about all of the Cisco products installed in the networking device that are assigned a PID, VID, and SN. If a Cisco entity is not assigned a PID, that entity is not retrieved or displayed.

Example:

```
Router# show inventory
NAME: “Chassis”, DESCR: “12008/GRP chassis”
PID: GSR8/40           ,  VID: V01,  SN: 63915640
NAME: “slot 0”, DESCR: “GRP”
PID: GRP-B             ,  VID: V01,  SN: CAB021300R5
```

```
NAME: “slot 1”, DESCR: “4 port ATM OC3 multimode”
PID: 4OC3/ATM-MM-SC    ,  VID: V01,  SN: CAB04036GT1
NAME: “slot 3”, DESCR: “4 port 0C3 POS multimode”
PID: LC-4OC3/POS-MM    ,  VID: V01,  SN: CAB014900GU
NAME: “slot 5”, DESCR: “1 port Gigabit Ethernet”
PID: GE-GBIC-SC-B      ,  VID: V01,  SN: CAB034251NX
NAME: “slot 7”, DESCR: “GRP”
PID: GRP-B             ,  VID: V01,  SN: CAB0428AN4O
NAME: “slot 16”, DESCR: “GSR 12008 Clock Scheduler Card”
PID: GSR8-CSC/ALRM     ,  VID: V01,  SN: CAB0429AUYH
NAME: “sfslot 1”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0428ALOS
NAME: “sfslot 2”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0429AU0M
NAME: “sfslot 3”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0429ARD7
NAME: “PSslot 1”, DESCR: “GSR 12008 AC Power Supply”
PID: FWR-GSR8-AC-B     ,  VID: V01,  SN: CAB041999CW
```

Enter the**showinventory**command with an *entity* argument value to display the UDI information for a specific type of Cisco entity installed in the networking device. In this example, a list of Cisco entities that match the sfslot argument string is displayed.

Example:

```
Router# show inventory sfslot
NAME: “sfslot 1”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0428ALOS
NAME: “sfslot 2”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0429AU0M
NAME: “sfslot 3”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0429ARD7
```

You can request even more specific UDI information using the**showinventory**command with an *entity* argument value that is enclosed in quotation marks. In this example, only the details for the entity that exactly matches the sfslot 1 argument string are displayed.

Example:

```
Router# show inventory “sfslot 1”
NAME: “sfslot 1”, DESCR: “GSR 12008 Switch Fabric Card”
PID: GSR8-SFC          ,  VID: V01,  SN: CAB0428ALOS
```

For diagnostic purposes, the**showinventory**command can be used with the **raw** keyword to display every RFC 2737 entity including those without a PID, UDI, or other physical identification.

Note The **raw** keyword option is primarily intended for troubleshooting problems with the **showinventory** command itself.

Example:

```
Router# show inventory raw
NAME: “Chassis”, DESCR: “12008/GRP chassis”
PID:                   ,  VID: V01,  SN: 63915640
NAME: “slot 0”, DESCR: “GRP”
PID:                   ,  VID: V01,  SN: CAB021300R5
NAME: “slot 1”, DESCR: “4 port ATM OC3 multimode”
PID: 4OC3/ATM-MM-SC    ,  VID: V01,  SN: CAB04036GT1
NAME: “slot 3”, DESCR: “4 port 0C3 POS multimode”
PID: LC-4OC3/POS-MM    ,  VID: V01,  SN: CAB014900GU
```

Enter the**showinventory**command with the **raw** keyword and an *entity* argument value to display the UDI information for the Cisco entities that are installed in the networking device and that match the argument string, even if they do not contain an assigned PID.

Example:

```
Router# show inventory raw slot
NAME: “slot 0”, DESCR: “GRP”
PID:                   ,  VID: V01,  SN: CAB021300R5
NAME: “slot 1”, DESCR: “4 port ATM OC3 multimode”
PID: 4OC3/ATM-MM-SC    ,  VID: V01,  SN: CAB04036GT1
NAME: “slot 3”, DESCR: “4 port 0C3 POS multimode”
PID: LC-4OC3/POS-MM    ,  VID: V01,  SN: CAB014900GU
```

Troubleshooting Tips

If any of the Cisco products do not have an assigned PID, the output may display incorrect PIDs and the VID and SN elements may be missing, as in the following example.

```
NAME: "Four Port High-Speed Serial", DESCR: "Four Port High-Speed Serial"
PID: Four Port High-Speed Serial, VID: 1.1, SN: 17202570
NAME: "Serial1/0", DESCR: "M4T"
PID: M4T               , VID:    , SN:
```

In the sample output, the PID is exactly the same as the product description. The UDI is designed for use with new Cisco products that have a PID assigned. UDI information on older Cisco products is not always reliable.

Configuration Examples for Unique Device Identifier Retrieval

There are no configuration examples for the UDI Retrieval feature. For sample display output from the show inventory command, see the “Retrieving the Unique Device Identifier” section.

Additional References

This section provides references related to the UDI Retrieval feature.

Related Documents

Related Topic	Document Title
Information about managing configuration files	• *Cisco IOS Configuration Fundamentals Configuration Guide*, Release 12.0 • *Cisco IOS Configuration Fundamentals Configuration Guide*, Release 12.2 • *Cisco IOS Configuration Fundamentals and Network Management Configuration Guide*, Release 12.3
Commands for showing interface statistics	• *Cisco IOS Interface Command Reference*, Release 12.0 • *Cisco IOS Interface Command Reference*, Release 12.2 • *Cisco IOS Interface and Hardware Component Command Reference*, Release 12.3T

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	--

MIBs

MIBs	MIBs Link
CISCO-ENTITY-ASSET-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2737	*Entity MIB (Version 2)*

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport