



Configuration Fundamentals Commands

activation-character

To define the character you enter at a vacant terminal to begin a terminal session, use the **activation-character** command in line configuration mode. To make any character activate a terminal, use the **no** form of this command.

activation-character *ascii-number*

no activation-character

Syntax Description

<i>ascii-number</i>	Decimal representation of the activation character.
---------------------	---

Defaults

Return (decimal 13)

Command Modes

Line configuration (config-line)

Command History

Release	Modification
10.0	This command was introduced.
	This command is supported in all Cisco IOS software Releases.

Usage Guidelines

See the “ASCII Character Set and Hexadecimal Values” document for a list of ASCII characters.



Note

If you are using the **autoselect** function, set the activation character to the default, Return, and exec-character-bits to 7. If you change these defaults, the application will not recognize the activation request.

Examples

The following example shows how to set the activation character for the console to Delete, which is decimal character 127:

```
Router(config)# line console
Router(config-line)# activation-character 127
```

alias

To create a command alias, use the **alias** command in global configuration mode. To delete all aliases in a command mode or to delete a specific alias, and to revert to the original command syntax, use the **no** form of this command.

alias *mode command-alias original-command*

no alias *mode [command-alias]*

Syntax Description

<i>mode</i>	Command mode of the original and alias commands.
<i>command-alias</i>	Command alias.
<i>original-command</i>	Original command syntax.

Defaults

A set of six basic EXEC mode aliases are enabled by default. See the “Usage Guidelines” section of this command for a list of default aliases.

Command Modes

Global configuration

Command History

Release	Modification
10.3	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

You can use simple words or abbreviations as command aliases.

[Table 11](#) lists the basic EXEC mode aliases that are enabled by default.

Table 11 **Default Command Aliases**

Command Alias	Original Command
h	help
lo	logout
p	ping
r	resume
s	show
w	where

The default aliases in [Table 11](#) are predefined. These default aliases can be disabled with the **no alias exec** command.

Common keyword aliases (which cannot be disabled) include **running-config** (keyword alias for **system:running-config**) and **startup-config** (keyword alias for **nvram:startup-config**). See the description of the **copy** command for more information about these keyword aliases.

Note that aliases can be configured for keywords instead of entire commands. You can create, for example, an alias for the first part of any command and still enter the additional keywords and arguments as normal.

To determine the value for the mode argument, enter the command mode in which you would issue the original command (and in which you will issue the alias) and enter the **?** command. The name of the command mode should appear at the top of the list of commands. For example, the second line in the following sample output shows the name of the command mode as “Interface configuration”:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# interface e0
Router(config-if)# ?
Interface configuration commands:
  access-expression      Build a bridge boolean access expression
  .
  .
  .
```

To match the name of the command mode to the acceptable mode keyword for the **alias** command, issue the **alias ?** command. As shown in the following sample output, the keyword needed to create a command alias for the access-expression command is **interface**:

```
Router(config)# alias ?
  accept-dialin          VPDN group accept dialin configuration mode
  accept-dialout         VPDN group accept dialout configuration mode
  address-family         Address Family configuration mode
  call-discriminator     Call Discriminator Configuration
  cascustom              Cas custom configuration mode
  clid-group             CLID group configuration mode
  configure              Global configuration mode
  congestion             Frame Relay congestion configuration mode
  controller            Controller configuration mode
  cptone-set            custom call progress tone configuration mode
  customer-profile       customer profile configuration mode
  dhcp                  DHCP pool configuration mode
  dnis-group            DNIS group configuration mode
  exec                  Exec mode
  flow-cache            Flow aggregation cache config mode
  fr-fr                FR/FR connection configuration mode
  interface              Interface configuration mode
  .
  .
  .
Router(config)# alias interface express access-expression
```

When you use online help, command aliases are indicated by an asterisk (*), and displayed in the following format:

**command-alias=original-command*

For example, the **lo** command alias is shown here along with other EXEC mode commands that start with “lo”:

```
Router# lo?
*lo=logout  lock  login  logout
```

When you use online help, aliases that contain multiple keyword elements separated by spaces are displayed in quotes, as shown here:

```
Router(config)#alias exec device-mail telnet device.cisco.com 25
Router(config)#end
Router#device-mail?
*device-mail="telnet device.cisco.com 25"
```

To list only commands and omit aliases, begin your input line with a space. In the following example, the alias **td** is not shown, because there is a space before the **t?** command line.

```
Router(config)#alias exec td telnet device
Router(config)#end
Router# t?
telnet terminal test tn3270 trace
```

To circumvent command aliases, use a space before entering the command. In the following example, the command alias **express** is not recognized because a space is used before the command.

```
Router(config-if)#exp?
*express=access-expression
Router(config-if)# express ?
% Unrecognized command
```

As with commands, you can use online help to display the arguments and keywords that can follow a command alias. In the following example, the alias **td** is created to represent the command **telnet device**. The **/debug** and **/line** switches can be added to **telnet device** to modify the command:

```
Router(config)#alias exec td telnet device
Router(config)#end
Router#td ?
    /debug      Enable telnet debugging mode
    /line       Enable telnet line mode
    ...
    whois       Whois port
    <cr>
Router# telnet device
```

You must enter the complete syntax for the command alias. Partial syntax for aliases is not accepted. In the following example, the parser does not recognize the command **t** as indicating the alias **td**:

```
Router# t
% Ambiguous command: "t"
```

Examples

In the following example, the alias **fixmyrt** is configured for the **clear iproute 192.168.116.16 EXEC** mode command:

```
Router(config)#alias exec fixmyrt clear ip route 192.168.116.16
```

In the following example, the alias **express** is configured for the first part of the **access-expression** interface configuration command:

```
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface e0
Router(config-if)#?
Interface configuration commands:
    access-expression      Build a bridge boolean access expression
    .
    .
    .
Router(config-if)#exit
Router(config)#alias ?
    accept-dialin          VPDN group accept dialin configuration mode
```

```

accept-dialout      VPDN group accept dialout configuration mode
address-family      Address Family configuration mode
call-discriminator  Call Discriminator Configuration
cascustom           Cas custom configuration mode
clid-group          CLID group configuration mode
configure           Global configuration mode
congestion          Frame Relay congestion configuration mode
controller          Controller configuration mode
cptone-set          custom call progress tone configuration mode
customer-profile    customer profile configuration mode
dhcp               DHCP pool configuration mode
dnis-group          DNIS group configuration mode
exec               Exec mode
flow-cache          Flow aggregation cache config mode
fr-fr              FR/FR connection configuration mode
interface           Interface configuration mode
.
.
.

```

```

Router(config)#alias interface express access-expression
Router(config)#int e0
Router(config-if)#exp?
*express=access-expression

```

```

Router(config-if)#express ?
  input  Filter input packets
  output Filter output packets

```

!Note that the true form of the command/keyword alias appears on the screen after issuing !the express ? command.

```

Router(config-if)#access-expression ?
  input  Filter input packets
  output Filter output packets
Router(config-if)#ex?
*express=access-expression exit

```

!Note that in the following line, a space is used before the ex? command !so the alias is not displayed.

```

Router(config-if)# ex?
exit

```

!Note that in the following line, the alias cannot be recognized because !a space is used before the command.

```

Router#(config-if)# express ?
% Unrecognized command

```

```

Router(config-if)# end
Router# show alias interface
Interface configuration mode aliases:
  express          access-expression

```

Related Commands

Command	Description
show aliases	Displays command aliases.

archive

To enter archive configuration mode, use the **archive** command in global configuration mode.

archive

Syntax Description This command has no arguments or keywords.

Command Default None

Command Modes Global configuration

Command History	Release	Modification
	12.3(4)T	This command was introduced.
	12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
	12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
	12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB and implemented on the Cisco 10000 series.

Examples The following example shows how to place the router in archive configuration mode:

```
Router# configure terminal
!
Router(config)# archive
Router(config-archive)#
```

Related Commands	Command	Description
	log config	Enters configuration change logger configuration mode.
	logging enable	Enables the logging of configuration changes.
	maximum	Sets the maximum number of archive files of the running configuration to be saved in the Cisco IOS configuration archive.
	path	Specifies the location and filename prefix for the files in the Cisco IOS configuration archive.
	time-period	Sets the time increment for automatically saving an archive file of the current running configuration in the Cisco IOS configuration archive.

archive config

To save a copy of the current running configuration to the Cisco IOS configuration archive, use the **archive config** command in privileged EXEC mode.

archive config

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(7)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(31)SB2	This command was implemented on the Cisco 10000 series.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB and implemented on the Cisco 10000 series.

Usage Guidelines



Note

Before using this command, you must configure the **path** command in order to specify the location and filename prefix for the files in the Cisco IOS configuration archive.

The Cisco IOS configuration archive is intended to provide a mechanism to store, organize, and manage an archive of Cisco IOS configuration files to enhance the configuration rollback capability provided by the **configure replace** command. Before this feature was introduced, you could save copies of the running configuration using the **copy running-config destination-url** command, storing the target file either locally or remotely. However, this method lacked any automated file management. On the other hand, the Configuration Replace and Configuration Rollback feature provides the capability to automatically save copies of the running configuration to the Cisco IOS configuration archive. These archived files serve as checkpoint configuration references and can be used by the **configure replace** command to revert to previous configuration states.

The **archive config** command allows you to save Cisco IOS configurations in the configuration archive using a standard location and filename prefix that is automatically appended with an incremental version number (and optional time stamp) as each consecutive file is saved. This functionality provides a means for consistent identification of saved Cisco IOS configuration files. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files has been saved in the archive, the oldest file is automatically deleted when the next, most recent file is saved. The **show archive** command displays information for all configuration files saved in the Cisco IOS configuration archive.

Examples

The following example shows how to save the current running configuration to the Cisco IOS configuration archive using the **archive config** command. Before using the **archive config** command, you must configure the **path** command to specify the location and filename prefix for the files in the Cisco IOS configuration archive. In this example, the location and filename prefix are specified as disk0:myconfig as follows:

```
configure terminal
!
archive
 path disk0:myconfig
end
```

You then save the current running configuration in the configuration archive, as follows:

```
archive config
```

The **show archive** command displays information on the files saved in the configuration archive as shown in the following sample output:

```
Router# show archive
```

```
There are currently 1 archive configurations saved.
The next archive file will be named disk0:myconfig-2
Archive #  Name
0
1      disk0:myconfig-1 <- Most Recent
2
3
4
5
6
7
8
9
10
```

Related Commands

Command	Description
archive	Enters archive configuration mode.
configure confirm	Confirms replacement of the current running configuration with a saved Cisco IOS configuration file.
configure replace	Replaces the current running configuration with a saved Cisco IOS configuration file.
maximum	Sets the maximum number of archive files of the running configuration to be saved in the Cisco IOS configuration archive.
path	Specifies the location and filename prefix for the files in the Cisco IOS configuration archive.
show archive	Displays information about the files saved in the Cisco IOS configuration archive.
time-period	Sets the time increment for automatically saving an archive file of the current running configuration in the Cisco IOS configuration archive.

archive log config persistent save

To save the persisted commands in the configuration log to the Cisco IOS secure file system, use the **archive log config persistent save** command in virtually any configuration mode.

archive log config persistent save

Syntax Description This command has no arguments or keywords.

Command Default If this command is not entered, the persisted configuration commands in the archive log are not saved to the Cisco IOS secure file system.

Command Modes Configuration change logger configuration mode in archive configuration mode is common for this command, but the command can be used in virtually any configuration mode.

Command History	Release	Modification
	12.2(33)SRA	This command was introduced.
	12.4(11)T	This command was integrated into Cisco IOS Release 12.4(11)T.
	12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
	12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Usage Guidelines This command can be entered at any time, regardless of what mode the router is in. If the router is in the persistent periodic mode, the persistent timer is restarted.

Examples The following example saves the persisted commands in the archive log to the Cisco IOS secure file system:

```
Router(config-archive-log-cfg)# archive log config persistent save
```

Related Commands	Command	Description
	log config	Enters configuration change logger configuration mode.
	logging enable	Enables the logging of configuration changes.
	logging persistent	Enables the configuration logging persistent feature.

archive tar

To create a TAR file, to list files in a TAR file, or to extract the files from a TAR file, use the **archive tar** command in privileged EXEC mode.

```
archive tar {/create destination-url flash:/file-url \ /table source-url \ /xtract source-url
flash:/file-url [dir/file...]}
```

Syntax Description

/create <i>destination-url</i> flash: / <i>file-url</i>	<p>Creates a new TAR file on the local or network file system.</p> <p>For <i>destination-url</i>, specify the destination URL alias for the local or network file system and the name of the TAR file to create. The following options are supported:</p> <ul style="list-style-type: none"> • flash:—Syntax for the local flash file system. • ftp:[[//<i>username[:password]</i>@<i>location</i>]/<i>directory</i>]/<i>tar-filename.tar</i>—Syntax for FTP. • rcp:[[//<i>username@location</i>]/<i>directory</i>]/<i>tar-filename.tar</i>—Syntax for Remote Copy Protocol (RCP). • tftp:[[//<i>location</i>]/<i>directory</i>]/<i>tar-filename.tar</i> —Syntax for TFTP. <p>The <i>tar-filename.tar</i> is the name of the TAR file to be created.</p> <p>For flash:/<i>file-url</i>, specify the location on the local flash file system from which the new TAR file is created.</p> <p>An optional list of files or directories within the source directory can be specified to write to the new TAR file. If none is specified, all files and directories at this level are written to the newly created TAR file.</p>
/table <i>source-url</i>	<p>Display the contents of an existing TAR file to the screen.</p> <p>For <i>source-url</i>, specify the source URL alias for the local or network file system. The following options are supported:</p> <ul style="list-style-type: none"> • flash:—Syntax for the local flash file system. • ftp:[[//<i>username[:password]</i>@<i>location</i>]/<i>directory</i>]/<i>tar-filename.tar</i>—Syntax for FTP. • rcp:[[//<i>username@location</i>]/<i>directory</i>]/<i>tar-filename.tar</i>—Syntax for Remote Copy Protocol (RCP). • tftp:[[//<i>location</i>]/<i>directory</i>]/<i>tar-filename.tar</i> —Syntax for TFTP. <p>The <i>tar-filename.tar</i> is the name of the TAR file to be created.</p>

/xtract <i>source-url</i> flash: <i>file-url</i> <i>[dir/file...]</i>	<p>Extracts files from a TAR file to the local file system.</p> <p>For <i>source-url</i>, specify the source URL alias for the local file system. These options are supported:</p> <ul style="list-style-type: none"> • flash:—Syntax for the local flash file system. • ftp:<i>[[/username[:password]@location]/directory]/tar-filename.tar</i>—Syntax for FTP. • rcp:<i>[[/username@location]/directory]/tar-filename.tar</i>—Syntax for Remote Copy Protocol (RCP). • tftp:<i>[[/location]/directory]/tar-filename.tar</i> —Syntax for TFTP. <p>The <i>tar-filename.tar</i> is the name of the TAR file to be created.</p>
---	---

Command Default The TAR archive file is not created.

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	12.1(13)AY	This command was introduced.
	12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
	12.4(22)YB	This command was integrated into Cisco IOS Release 12.4(22)YB.
	12.4(24)T	This command was integrated into Cisco IOS Release 12.4(24)T.

Usage Guidelines Filenames, directory names, and image names are case sensitive.

The TAR file is an archive file from which you can extract files by using the **archive tar** command.

Examples The following example shows how to create a TAR file. The command writes the contents of the new-configs directory on the local flash device to a file named saved.tar on the TFTP server at 172.20.136.9.

```
Switch# archive tar /create tftp:172.20.136.9/saved.tar flash:/new-configs
```

The following example shows how to display the contents of the c2940-tv0-m.tar file that is in flash memory. The contents of the TAR file appear on the screen.

```
Switch# archive tar /table flash:c2940-tv0-m.tar

info (219 bytes)
c2940-tv0-mz-121/ (directory)
c2940-tv0-mz-121/html/ (directory)
c2940-tv0-mz-121/html/foo.html (0 bytes)
c2940-tv0-mz-121/vegas-tv0-mz-121.bin (610856 bytes)
c2940-tv0-mz-121/info (219 bytes)
info.ver (219 bytes)
```

The following example shows how to extract the contents of a TAR file on the TFTP server at 172.20.10.30. This command extracts only the new-configs directory into the root directory on the local flash file system. The remaining files in the saved.tar file are ignored.

```
Switch# archive tar /xtract tftp://172.20.10.30/saved.tar flash:/ new-configs
```

async-bootp

To configure extended BOOTP requests for asynchronous interfaces as defined in RFC 1084, use the **async-bootp** command in global configuration mode. To restore the default, use the **no** form of this command.

async-bootp *tag* [:*hostname*] *data*

no async-bootp

Syntax Description

<i>tag</i>	Item being requested; expressed as filename, integer, or IP dotted decimal address. See Table 12 for possible keywords.
: <i>hostname</i>	(Optional) This entry applies only to the specified host. The <i>:hostname</i> argument accepts both an IP address and a logical host name.
<i>data</i>	List of IP addresses entered in dotted decimal notation or as logical host names, a number, or a quoted string.

Table 12 *tag* Keyword Options

Keyword	Description
bootfile	Specifies use of a server boot file from which to download the boot program. Use the optional <i>:hostname</i> argument and the <i>data</i> argument to specify the filename.
subnet-mask <i>mask</i>	Dotted decimal address specifying the network and local subnetwork mask (as defined by RFC 950).
time-offset <i>offset</i>	Signed 32-bit integer specifying the time offset of the local subnetwork in seconds from Coordinated Universal Time (UTC).
gateway <i>address</i>	Dotted decimal address specifying the IP addresses of gateways for this subnetwork. A preferred gateway should be listed first.
time-server <i>address</i>	Dotted decimal address specifying the IP address of time servers (as defined by RFC 868).
IEN116-server <i>address</i>	Dotted decimal address specifying the IP address of name servers (as defined by IEN 116).
nbns-server <i>address</i>	Dotted decimal address specifying the IP address of Windows NT servers.
DNS-server <i>address</i>	Dotted decimal address specifying the IP address of domain name servers (as defined by RFC 1034).
log-server <i>address</i>	Dotted decimal address specifying the IP address of an MIT-LCS UDP log server.
quote-server <i>address</i>	Dotted decimal address specifying the IP address of Quote of the Day servers (as defined in RFC 865).
lpr-server <i>address</i>	Dotted decimal address specifying the IP address of Berkeley UNIX Version 4 BSD servers.
impress-server <i>address</i>	Dotted decimal address specifying the IP address of Impress network image servers.

Table 12 tag Keyword Options (continued)

Keyword	Description
rlp-server <i>address</i>	Dotted decimal address specifying the IP address of Resource Location Protocol (RLP) servers (as defined in RFC 887).
hostname <i>name</i>	The name of the client, which may or may not be domain qualified, depending upon the site.
bootfile-size <i>value</i>	A two-octet value specifying the number of 512-octet (byte) blocks in the default boot file.

Defaults

If no extended BOOTP commands are entered, the Cisco IOS software generates a gateway and subnet mask appropriate for the local network.

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Use the **show async-bootp EXEC** command to list the configured parameters. Use the **no async-bootp** command to clear the list.

Examples

The following example illustrates how to specify different boot files: one for a PC, and one for a Macintosh. With this configuration, a BOOTP request from the host on 172.30.1.1 results in a reply listing the boot filename as `pcboot`. A BOOTP request from the host named "mac" results in a reply listing the boot filename as "macboot."

```
async-bootp bootfile :172.30.1.1 "pcboot"
async-bootp bootfile :mac "macboot"
```

The following example specifies a subnet mask of 255.255.0.0:

```
async-bootp subnet-mask 255.255.0.0
```

The following example specifies a negative time offset of the local subnetwork of 3600 seconds:

```
async-bootp time-offset -3600
```

The following example specifies the IP address of a time server:

```
async-bootp time-server 172.16.1.1
```

Related Commands

Command	Description
show async bootp	Displays the extended BOOTP request parameters that have been configured for asynchronous interfaces.

attach

To connect to a specific line card or module from a remote location for the purpose of executing monitoring and maintenance commands on that line card or module, use the **attach** command in privileged EXEC mode. To exit from the Cisco IOS software image on the line card and return to the Cisco IOS image on the main (Supervisor) module, use the **exit** command.

Cisco 12000 Series

attach *slot-number*

Cisco 7600 Series and Catalyst 6500 Series

attach *module-number*

Syntax Description

<i>slot-number</i>	Slot number of the line card to which you wish to connect. If you omit the slot number, you will be prompted for it.
<i>module-number</i>	Module number; see the “Usage Guidelines” section for valid values.

Defaults

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
11.2GS	This command was introduced on the Cisco 12000 series.
12.2(14)SX	This command was implemented on the Supervisor Engine 720.
12.2(17d)SXB	Support was added for the Supervisor Engine 2.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Cisco 12000 Series

You must first use the **attach** privileged EXEC command to access the Cisco IOS software image on a line card before using line card-specific **show** EXEC commands. Alternatively, you can use the **execute-on** privileged EXEC command to execute a **show** command on a specific line card.

After you connect to the Cisco IOS image on the line card using the **attach** command, the prompt changes to `LC-Slotx#`, where *x* is the slot number of the line card.

The commands executed on the line card use the Cisco IOS image on that line card.

You can also use the **execute-on slot** privileged EXEC command to execute commands on one or all line cards.



Note

Do not execute the **config** EXEC command from the Cisco IOS software image on the line card.

Cisco 7600 Series and Catalyst 6500 Series



Caution

After you enter the **attach** or **remote login** command to access another console from your switch, if you enter global or interface configuration mode commands, the switch might reset.

The valid values for the *module-number* argument depend on the chassis that is used. For example, if you have a 13-slot chassis, valid values for the module number are from 1 to 13.

This command is supported on Distributed Forwarding Card (DFC)-equipped modules, FlexWan modules, and the supervisor engine only.

When you execute the **attach module-number** command, the prompt changes to Router-dfcx# or Switch-sp#, depending on the type of module to which you are connecting.

The behavior of the **attach** command is identical to that of the **remote login module num** command.

There are two ways to end this session:

- You can enter the **exit** command as follows:

```
Router-dfc3# exit

[Connection to Switch closed by foreign host]
Router#
```

- You can press **Ctrl-C** three times as follows:

```
Router-dfc3# ^C
Router-dfc3# ^C
Router-dfc3# ^C
Terminate remote login session? [confirm] y
[Connection to Switch closed by local host]
Router#
```

Examples

In the following example, the user connects to the Cisco IOS image running on the line card in slot 9, gets a list of valid **show** commands, and returns the Cisco IOS image running on the GRP:

```
Router# attach 9

Entering Console for 4 Port Packet Over SONET OC-3c/STM-1 in Slot: 9
Type exit to end this session

Press RETURN to get started!

LC-Slot9# show ?

 cef          Cisco Express Forwarding
 clock        Display the system clock
 context      Show context information about recent crash(s)
 history      Display the session command history
 hosts        IP domain-name, lookup style, nameservers, and host table
 ipc          Interprocess communications commands
 location     Display the system location
 sessions     Information about Telnet connections
 terminal     Display terminal configuration parameters
 users        Display information about terminal lines
 version      System hardware and software status

LC-Slot9# exit

Disconnecting from slot 9.
```

```
Connection Duration: 00:01:04
Router#
```

**Note**

Because not all statistics are maintained on line cards, the output from some of **show** commands may be inconsistent.

The following example shows how to log in remotely to the DFC-equipped module:

```
Console# attach 3

Trying Switch ...
Entering CONSOLE for Switch
Type "^C^C^C" to end this session

Router-dfc3#
```

Related Commands

Command	Description
attach shelf	Connects you to a specific (managed) shelf for the purpose of remotely executing commands on that shelf only.
execute-on slot	Executes commands remotely on a specific line card, or on all line cards simultaneously.
remote login	Accesses the Cisco 7600 series router console or a specific module.

autobaud

To set the line for automatic baud rate detection (autobaud), use the **autobaud** command in line configuration mode. To disable automatic baud detection, use the **no** form of this command.

autobaud

no autobaud

Syntax Description

This command has no arguments or keywords.

Defaults

Autobaud detection is disabled. Fixed speed of 9600 bps.

Command Modes

Line configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The autobaud detection supports a range from 300 to 19200 baud. A line set for autobaud cannot be used for outgoing connections, nor can you set autobaud capability on a line using 19200 baud when the parity bit is set (because of hardware limitations).



Note

Automatic baud detection must be disabled by using the **no autobaud** command prior to setting the **txspeed**, **rxspeed**, or **speed** commands.

Examples

In the following example, the auxiliary port is configured for autobaud detection:

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

auto-sync

To enable automatic synchronization of the configuration files in NVRAM, use the **auto-sync** command in main-cpu redundancy configuration mode. To disable automatic synchronization, use the **no** form of this command.

auto-sync { **startup-config** | **config-register** | **bootvar** | **running-config** | **standard** }

no auto-sync { **startup-config** | **config-register** | **bootvar** | **standard** }

Syntax Description

startup-config	Specifies synchronization of the startup configuration files.
config-register	Specifies synchronization of the configuration register values.
bootvar	Specifies synchronization of the following boot variables: <ul style="list-style-type: none"> • BOOT—Set by the boot system <i>device:filename</i> command. • CONFIG_FILE—Set by the boot config <i>device:filename</i> command. • BOOTLDR—Set by the boot bootldr <i>device:filename</i> command.
running-config	Specifies synchronization of the running configuration files.
standard	Specifies synchronization of all of the system files (startup configuration, boot variables, and config configuration registers).

Defaults

For the Performance Routing Engines (PREs) on the Cisco uBR10012 universal broadband router, the system defaults to synchronizing all system files on the (**auto-sync standard**).

For the Supervisor Engines on the Cisco 7600 series routers, the system defaults to synchronizing the running configuration. (**running-config**).

Command Modes

Main-cpu redundancy configuration

Command History

Release	Modification
12.2(4)XF1	This command was introduced on the Cisco uBR10012 universal broadband router.
12.2(14)SX	This command was integrated into the Supervisor Engine 720.
12.2(17d)SXB	Support was added for the Supervisor Engine 2.
12.2(18)SXD	Support for this command on the Cisco 7600 series routers was removed.
12.3BC	This command was integrated into Cisco IOS Release 12.3BC for the Cisco uBR10012 router.
12.2(33)SCA	This command is obsolete on the Cisco uBR10012 universal broadband router.

Usage Guidelines

Cisco 7600 Series Routers

If you enter the **no auto-sync standard** command, no automatic synchronizations occur. If you want to enable any of the keywords, you have to enter the appropriate command for each keyword.

The **auto-sync** command is not supported in RPR+ mode.

Cisco uBR10012 Universal Broadband Router

By default, the system synchronizes all system files, which is the typical setting for most applications. However, you might want exclude certain files from synchronization for specialized applications.

For example, if you have configured the active and standby PRE1 (or PRE2) modules to run different versions of Cisco IOS software, you might want to use different configuration files as well. In this case, you would not synchronize the startup configuration file.

Examples**Cisco 7600 Series Routers**

The following example shows how (from the default configuration) to enable automatic synchronization of the configuration register in the main CPU:

```
Router# configure terminal
Router (config)# redundancy
Router (config-r)# main-cpu
Router (config-r-mc)# no auto-sync standard
Router (config-r-mc)# auto-sync config-register
```

Cisco uBR10012 Universal Broadband Router

The following example shows the system being configured to synchronize only the startup configuration file:

```
router(config)# redundancy
router(config-r)# main-cpu
router(config-r-mc)# auto-sync startup-config
router(config-r-mc)# exit
router(config-r)# exit
```

The following example shows how to configure the system to synchronize all system files except for the startup configuration file. This typically is done when the two PRE1 (or PRE2) modules are running different software images.

```
router(config)# redundancy
router(config-r)# main-cpu
router(config-r-mc)# no auto-sync startup-config
router(config-r-mc)# auto-sync config-register
router(config-r-mc)# auto-sync bootvar
router(config-r-mc)# exit
router(config-r)# exit
```

Related Commands

Command	Description
redundancy	Enters redundancy configuration mode.
main-cpu	Enters main CPU redundancy configuration mode.

autoupgrade disk-cleanup

To configure the Cisco IOS Auto-Upgrade Manager disk cleanup utility, use the **autoupgrade disk-cleanup** command in global configuration mode. To disable this configuration, use the **no** form of this command.

autoupgrade disk-cleanup [**crashinfo** | **core** | **image** | **irrecoverable**]

no autoupgrade disk-cleanup [**crashinfo** | **core** | **image** | **irrecoverable**]

Syntax Description

crashinfo	(Optional) Deletes crashinfo files during disk-cleanup before an image is downloaded.
core	(Optional) Deletes core files during disk-cleanup before an image is downloaded.
image	(Optional) Deletes the Cisco IOS images, except the default boot image and the current image, during disk-cleanup before an image is downloaded.
irrecoverable	(Optional) Deletes files irretrievably (in a file-system that supports the undelete operation) during disk-cleanup before an image is downloaded.

Command Default

By default, the crashinfo files, the core files, and the Cisco IOS software images are deleted by the Cisco IOS Auto-Upgrade Manager disk cleanup utility, and the filesystems that support the undelete operation are not cleaned up.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(15)T	This command was introduced.

Examples

The following example shows how to clean-up filesystems that support undelete operation:

```
Router(config)# autoupgrade disk-cleanup irrecoverable
```

The following example shows how to avoid deleting the Cisco IOS software images:

```
Router(config)# no autoupgrade disk-cleanup image
```

Related Commands

Command	Description
autoupgrade ida url	Configures the URL of the server on www.cisco.com where the image download requests will be sent by Auto-Upgrade Manager.
autoupgrade status email	Configures the address to which the status email is to be sent.
upgrade automatic getversion	Downloads a Cisco IOS software image directly from www.cisco.com or from a non-Cisco server.

autoupgrade ida url

To configure the URL of the Intelligent Download Application (IDA) running on www.cisco.com, use the **autoupgrade ida url** command in global configuration mode. The router will send the image download requests to the configured URL. To disable this URL, use the **no** form of this command.

autoupgrade ida url *url*

no autoupgrade ida url *url*

Syntax Description

<i>url</i>	URL of the IDA server.
------------	------------------------

Command Default

Default URL: https://www.cisco.com/cgi-bin/ida/locator/locator.pl

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(15)T	This command was introduced.

Usage Guidelines

Use the **autoupgrade ida url** command to configure a new URL for the IDA server, if it is not present in the default location.

Examples

The following example shows how to configure the URL for the IDA server:

```
Router(config)# autoupgrade ida url https://www.cisco.com/cgi-bin/ida/locator/locator.pl
```

Related Commands

Command	Description
autoupgrade disk-cleanup	Configures the Cisco IOS Auto-Upgrade Manager disk cleanup utility.
autoupgrade status email	Configures the address to which the status email is to be sent.
upgrade automatic getversion	Downloads a Cisco IOS software image directly from www.cisco.com or from a non-Cisco server.

autoupgrade status email

To configure the address to which status email is to be sent and the outgoing email server, use the **autoupgrade status email** command in global configuration mode. To disable status email, use the **no** form of this command.

autoupgrade status email [**recipient** *email-address*] [**smtp-server***smtp-server*]

no autoupgrade status email [**recipient** *email-address*] [**smtp-server***smtp-server*]

Syntax Description

recipient	The address to which the Cisco IOS Auto-Upgrade Manager (AUM) status is to be sent.
smtp-server	The outgoing email server to which the AUM email is sent.
<i>email-address</i>	The email address to which the AUM status is to be sent.

Command Default

Status email is not sent unless the address is configured. The recipient email address and SMTP server have to be configured in order to receive AUM status email.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(15)T	This command was introduced.

Usage Guidelines

Use this command to configure the email-address where AUM status email can be sent.

Examples

The following example shows how to configure the address to which status email is to be sent:

```
Router(config)# autoupgrade status email recipient tree@abc.com
```

```
Router(config)# autoupgrade status email smtp-server smtpserver.abc.com
```

Related Commands

Command	Description
autoupgrade disk-cleanup	Configures the Cisco IOS Auto-Upgrade Manager disk cleanup utility.
autoupgrade ida url	Configures the URL of the server running on www.cisco.com to which the router sends the image download requests.
upgrade automatic getversion	Downloads a Cisco IOS software image directly from www.cisco.com or from a non-Cisco server.

banner exec

To specify and enable a message to be displayed when an EXEC process is created (an EXEC banner), use the **banner exec** command in global configuration mode. To delete the existing EXEC banner, use the **no** form of this command.

```
banner exec d message d
```

```
no banner exec
```

Syntax Description

<i>d</i>	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
<i>message</i>	Message text. You can include tokens in the form $\$(token)$ in the message text. Tokens will be replaced with the corresponding configuration variable. Tokens are described in Table 13 .

Defaults

Disabled (no EXEC banner is displayed).

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
11.3(7.5)AA	Token functionality was introduced.
12.0(3)T	Token functionality was integrated into Cisco IOS Release 12.0(3)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

This command specifies a message to be displayed when an EXEC process is created (a line is activated, or an incoming connection is made to a vty). Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

When a user connects to a router, the message-of-the-day (MOTD) banner appears first, followed by the login banner and prompts. After the user logs in to the router, the EXEC banner or incoming banner will be displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner will be displayed. For all other connections, the router will display the EXEC banner.

To disable the EXEC banner on a particular line or lines, use the **no exec-banner** line configuration command.

To customize the banner, use tokens in the form $\$(token)$ in the message text. Tokens will display current Cisco IOS configuration variables, such as the router's host name and IP address. The tokens are described in [Table 13](#).

Table 13 *banner exec Tokens*

Token	Information Displayed in the Banner
\$(hostname)	Displays the host name for the router.
\$(domain)	Displays the domain name for the router.
\$(line)	Displays the vty or tty (asynchronous) line number.
\$(line-desc)	Displays the description attached to the line.

Examples

The following example sets an EXEC banner that uses tokens. The percent sign (%) is used as a delimiting character. Notice that the $$(token)$ syntax is replaced by the corresponding configuration variable.

```
Router(config)# banner exec %
Enter TEXT message. End with the character '%'.
Session activated on line $(line), $(line-desc). Enter commands at the prompt.
%
```

When a user logs on to the system, the following output is displayed:

```
User Access Verification

Username: joeuser
Password: <password>

Session activated on line 50, vty default line. Enter commands at the prompt.

Router>
```

Related Commands

Command	Description
banner incoming	Defines a customized banner to be displayed when there is an incoming connection to a terminal line from a host on the network.
banner login	Defines a customized banner to be displayed before the username and password login prompts.
banner motd	Defines a customized message-of-the-day banner.
banner slip-ppp	Defines a customized banner to be displayed when a serial-line IP or point-to-point connection is made.
exec-banner	Controls (enables or disables) the display of EXEC banners and message-of-the-day banners on a specified line or lines.

banner incoming

To define and enable a banner to be displayed when there is an incoming connection to a terminal line from a host on the network, use the **banner incoming** command in global configuration mode. To delete the incoming connection banner, use the **no** form of this command.

```
banner incoming d message d
```

```
no banner incoming
```

Syntax Description

<i>d</i>	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
<i>message</i>	Message text. You can include tokens in the form $\$(token)$ in the message text. Tokens will be replaced with the corresponding configuration variable. Tokens are described in Table 14 .

Defaults

Disabled (no incoming banner is displayed).

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
11.3(7.5)AA	Token functionality was introduced.
12.0(3)T	Token functionality was integrated into Cisco IOS Release 12.0(3)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Follow the **banner incoming** command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

An *incoming connection* is one initiated from the network side of the router. Incoming connections are also called reverse Telnet sessions. These sessions can display MOTD banners and incoming banners, but they do not display EXEC banners. Use the **no motd-banner** line configuration command to disable the MOTD banner for reverse Telnet sessions on asynchronous lines.

When a user connects to the router, the message-of-the-day (MOTD) banner (if configured) appears first, before the login prompt. After the user successfully logs in to the router, the EXEC banner or incoming banner will be displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner will be displayed. For all other connections, the router will display the EXEC banner.

Incoming banners cannot be suppressed. If you do not want the incoming banner to appear, you must delete it with the **no banner incoming** command.

To customize the banner, use tokens in the form $\$(token)$ in the message text. Tokens will display current Cisco IOS configuration variables, such as the router's host name and IP address. The tokens are described in [Table 14](#).

Table 14 *banner incoming Tokens*

Token	Information Displayed in the Banner
\$(hostname)	Displays the host name for the router.
\$(domain)	Displays the domain name for the router.
\$(line)	Displays the vty or tty (asynchronous) line number.
\$(line-desc)	Displays the description attached to the line.

Examples

The following example sets an incoming connection banner. The pound sign (#) is used as a delimiting character.

```
Router(config)# banner incoming #
This is the Reuses router.
#
```

The following example sets an incoming connection banner that uses several tokens. The percent sign (%) is used as a delimiting character.

```
darkstar(config)# banner incoming %
Enter TEXT message. End with the character '%'.
You have entered $(hostname).$(domain) on line $(line) ($(line-desc)) %
```

When the incoming connection banner is executed, the user will see the following banner. Notice that the *\$(token)* syntax is replaced by the corresponding configuration variable.

```
You have entered darkstar.ourdomain.com on line 5 (Dialin Modem)
```

Related Commands

Command	Description
banner exec	Defines a customized banner to be displayed whenever the EXEC process is initiated.
banner login	Defines a customized banner to be displayed before the username and password login prompts.
banner motd	Defines a customized message-of-the-day banner.
banner slip-ppp	Defines a customized banner to be displayed when a serial-line IP or point-to-point connection is made.

banner login

To define and enable a customized banner to be displayed before the username and password login prompts, use the **banner login** command in global configuration mode. To disable the login banner, use **no** form of this command.

banner login *d message d*

no banner login

Syntax Description

<i>d</i>	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
<i>message</i>	Message text. You can include tokens in the form $\$(token)$ in the message text. Tokens will be replaced with the corresponding configuration variable. Tokens are described in Table 15 .

Defaults

Disabled (no login banner is displayed).

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
11.3(7.5)AA	Token functionality was introduced.
12.0(3)T	Token functionality was integrated into Cisco IOS Release 12.0(3)T.
12.2(14)SX	This command was integrated into Cisco IOS Release 12.2(14)SX.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Follow the **banner login** command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

When a user connects to the router, the message-of-the-day (MOTD) banner (if configured) appears first, followed by the login banner and prompts. After the user successfully logs in to the router, the EXEC banner or incoming banner will be displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner will be displayed. For all other connections, the router will display the EXEC banner.

To customize the banner, use tokens in the form $\$(token)$ in the message text. Tokens will display current Cisco IOS configuration variables, such as the router's host name and IP address. The tokens are described in [Table 15](#).

Table 15 *banner login Tokens*

Token	Information Displayed in the Banner
\$(hostname)	Displays the host name for the router.
\$(domain)	Displays the domain name for the router.
\$(line)	Displays the vty or tty (asynchronous) line number.
\$(line-desc)	Displays the description attached to the line.

Examples

The following example sets a login banner. Double quotes (") are used as the delimiting character.

```
Router# banner login " Access for authorized users only. Please enter your username and password. "
```

The following example sets a login banner that uses several tokens. The percent sign (%) is used as the delimiting character.

```
darkstar(config)# banner login %
Enter TEXT message. End with the character '%'.
You have entered $(hostname).$(domain) on line $(line) ($(line-desc)) %
```

When the login banner is executed, the user will see the following banner. Notice that the \$(token) syntax is replaced by the corresponding configuration variable.

```
You have entered darkstar.ourdomain.com on line 5 (Dialin Modem)
```

Related Commands

Command	Description
banner exec	Defines a customized banner to be displayed whenever the EXEC process is initiated.
banner incoming	Defines a customized message to be displayed when there is an incoming connection to a terminal line from a host on the network.
banner motd	Defines a customized message-of-the-day banner.
banner slip-ppp	Defines a customized banner to be displayed when a serial-line IP or point-to-point connection is made.

banner motd

To define and enable a message-of-the-day (MOTD) banner, use the **banner motd** command in global configuration mode. To delete the MOTD banner, use the **no** form of this command.

```
banner motd d message d
```

```
no banner motd
```

Syntax Description

<i>d</i>	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
<i>message</i>	Message text. You can include tokens in the form $\$(token)$ in the message text. Tokens will be replaced with the corresponding configuration variable.

Defaults

Disabled (no MOTD banner is displayed).

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
11.3(7.5)AA	Token functionality was introduced.
12.0(3)T	Token functionality was integrated into Cisco IOS Release 12.0(3)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

This MOTD banner is displayed to all terminals connected and is useful for sending messages that affect all users (such as impending system shutdowns). Use the **no exec-banner** or **no motd-banner** command to disable the MOTD banner on a line. The **no exec-banner** command also disables the EXEC banner on the line.

When a user connects to the router, the MOTD banner appears before the login prompt. After the user logs in to the router, the EXEC banner or incoming banner will be displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner will be displayed. For all other connections, the router will display the EXEC banner.

To customize the banner, use tokens in the form $\$(token)$ in the message text. Tokens will display current Cisco IOS configuration variables, such as the router's host name and IP address. The tokens are described in [Table 16](#).

Table 16 *banner motd Tokens*

Token	Information Displayed in the Banner
\$(hostname)	Displays the host name for the router.
\$(domain)	Displays the domain name for the router.
\$(line)	Displays the vty or tty (asynchronous) line number.
\$(line-desc)	Displays the description attached to the line.

Examples

The following example configures an MOTD banner. The pound sign (#) is used as a delimiting character.

```
Router# banner motd # Building power will be off from 7:00 AM until 9:00 AM this coming
Tuesday. #
```

The following example configures an MOTD banner with a token. The percent sign (%) is used as a delimiting character.

```
darkstar(config)# banner motd %
Enter TEXT message. End with the character '%'.
Notice: all routers in $(domain) will be upgraded beginning April 20
%
```

When the MOTD banner is executed, the user will see the following. Notice that the *\$(token)* syntax is replaced by the corresponding configuration variable.

```
Notice: all routers in ourdomain.com will be upgraded beginning April 20
```

Related Commands

Command	Description
banner exec	Defines and enables a customized banner to be displayed whenever the EXEC process is initiated.
banner incoming	Defines and enables a customized message to be displayed when there is an incoming connection to a terminal line from a host on the network.
banner login	Defines and enables a customized banner to be displayed before the username and password login prompts.
banner slip-ppp	Defines and enables a customized banner to be displayed when a serial-line IP or point-to-point connection is made.
exec-banner	Controls (enables or disables) the display of EXEC banners and message-of-the-day banners on a specified line or lines.
motd-banner	Controls (enables or disables) the display of message-of-the-day banners on a specified line or lines.

banner slip-ppp

To customize the banner that is displayed when a Serial Line Internet Protocol (SLIP) or PPP connection is made, use the **banner slip-ppp** command in global configuration mode. To restore the default SLIP or PPP banner, use the **no** form of this command.

```
banner slip-ppp d message d
```

```
no banner slip-ppp
```

Syntax Description

<i>d</i>	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
<i>message</i>	Message text. You can include tokens in the form $\$(token)$ in the message text. Tokens will be replaced with the corresponding configuration variable.

Defaults

The default SLIP or PPP banner message is:

```
Entering encapsulation mode.  
Async interface address is unnumbered (Ethernet0)  
Your IP address is 10.000.0.0 MTU is 1500 bytes
```

The banner message when using the **service old-slip-prompt** command is:

```
Entering encapsulation mode.  
Your IP address is 10.100.0.0 MTU is 1500 bytes
```

where *encapsulation* is SLIP or PPP.

Command Modes

Global configuration

Command History

Release	Modification
12.0(3)T	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

Use this command to define a custom SLIP or PPP connection message. This is useful when legacy client applications require a specialized connection string. To customize the banner, use tokens in the form $\$(token)$ in the message text. Tokens will display current Cisco IOS configuration variables, such as the routers host name, IP address, encapsulation type, and Maximum Transfer Unit (MTU) size. The banner tokens are described in [Table 17](#).

Table 17 *banner slip-ppp Tokens*

Token	Information Displayed in the Banner
\$(hostname)	Displays the host name of the router.
\$(domain)	Displays the domain name of the router.
\$(peer-ip)	Displays the IP address of the peer machine.
\$(gate-ip)	Displays the IP address of the gateway machine.
\$(encap)	Displays the encapsulation type (SLIP, PPP, and so on).
\$(encap-alt)	Displays the encapsulation type as SL/IP instead of SLIP.
\$(mtu)	Displays the MTU size.

Examples

The following example sets the SLIP/PPP banner using several tokens and the percent sign (%) as the delimiting character:

```
Router(config)# banner slip-ppp %
Enter TEXT message. End with the character '%'.
Starting $(encap) connection from $(gate-ip) to $(peer-ip) using a maximum packet size of
$(mtu) bytes... %
```

The new SLIP/PPP banner will now be displayed when the **slip** EXEC command is used. Notice that the *\$(token)* syntax is replaced by the corresponding configuration variable.

```
Router# slip
Starting SLIP connection from 172.16.69.96 to 192.168.1.200 using a maximum packet size of
1500 bytes...
```

Related Commands

Command	Description
banner exec	Defines and enables a customized banner to be displayed whenever the EXEC process is initiated.
banner incoming	Defines and enables a customized message to be displayed when there is an incoming connection to a terminal line from a host on the network.
banner motd	Defines and enables a customized message-of-the-day banner.
ppp	Initiates a connection to a remote host using PPP.
slip	Initiates a connection to a remote host using SLIP.

boot

To boot the router manually, use the **boot** command in ROM monitor mode. The syntax of this command varies according to the platform and ROM monitor version.

boot

boot *file-url*

boot *filename* [*tftp-ip-address*]

boot flash [*flash-fs*][:*partition-number*][:*filename*]

Cisco 7000 Series, 7200 Series, 7500 Series Routers

boot *flash-fs*[:*filename*]

Cisco 1600 and Cisco 3600 Series Routers

boot [*flash-fs*][:*partition-number*][:*filename*]

Cisco 1800 Series, 2800 Series, and 3800 Series Routers

boot usbflash0[:*filename*]

Syntax Description

<i>file-url</i>	URL of the image to boot (for example, boot tftp://172.16.15.112/router-test).
<i>filename</i>	<p>When used in conjunction with the <i>ip-address</i> argument, the <i>filename</i> argument is the name of the system image file to boot from a network server. The filename is case sensitive.</p> <p>When used in conjunction with the flash keyword, the <i>filename</i> argument is the name of the system image file to boot from Flash memory.</p> <p>On all platforms except the Cisco 1600 series, Cisco 3600 series, and Cisco 7000 family routers, the system obtains the image file from internal Flash memory.</p> <p>On the Cisco 1600 series, Cisco 3600 series and Cisco 7000 family routers, the <i>flash-fs</i> argument specifies the Flash memory device from which to obtain the system image. (See the <i>flash-fs</i> argument later in this table for valid device values.) The filename is case sensitive. Without the <i>filename</i> argument, the first valid file in Flash memory is loaded.</p> <p>If the <i>filename</i> is not specified, the first file in the partition or file system is used. (A USB Flash uses the first image in (compact) Flash as the boot loader and loads the image from USB Flash.)</p>
<i>tftp-ip-address</i>	(optional) IP address of the TFTP server on which the system image resides. If omitted, this value defaults to the IP broadcast address of 255.255.255.255.

flash	Boots the router from Flash memory. Note that this keyword is required in some boot images.
usbflash0	Boot the first file in USB Flash 0. If the optional filename argument is used, the router boots the specified image from USB Flash. Note This option uses the first image in (compact) Flash as the boot loader and loads the image from USB Flash.
<i>flash-fs:</i>	(Optional) Specifying the Flash file system is optional for all platforms except the Cisco 7500 series routers. Possible file systems are: <ul style="list-style-type: none"> • flash:—Internal Flash memory. • bootflash:—Internal Flash memory on the Cisco 7000 family. • slot0:—Flash memory card in the first PCMCIA slot on the Cisco 7000 family and Cisco 3600 series routers. • slot1:—Flash memory card in the second PCMCIA slot on the Cisco 7000 family and Cisco 3600 series routers.
<i>partition-number:</i>	(Optional) Specifies the partition number of the file system the file should be loaded from. This argument is not available on all platforms. If the <i>partition-number</i> is not specified, the first partition is used.

Defaults

For most platforms, if you enter the **boot** command and press Enter, the router boots from ROM by default. However, for some platforms, such as the Cisco 3600 series routers, if you enter the **boot** command and press Enter, the router boots the first image in Flash memory. Refer to the documentation for your platform for information about the default image.

Command Modes

ROM monitor

Command History

Release	Modification
10.3	The command was introduced.
12.3(14)T	The usbflash0 keyword was added to support booting an image from an external USB Flash drive.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

To determine which form of this command to use, refer to the documentation for your platform or use the CLI help (?) feature.

Use this command only when your router cannot find the boot configuration information needed in NVRAM. To enter ROM monitor mode, use one of the following methods:

- Enter the **reload EXEC** command, then press the **Break** key during the first 60 seconds of startup.
- Set the configuration register bits 0 to 3 to zero (for example, set the configuration register to 0x0) and enter the **reload** command.


```
> boot flash flash:2:igs-bpx-1
F3: 3562264+98228+303632 at 0x30000B4
```

(ROM Monitor copyrights)

In the following command, the Cisco 7000 family router accepts the **flash** keyword for compatibility but ignores it, and boots from slot 0:

```
> boot flash slot0:gs7-k-mz.103-9
F3: 8468+3980384+165008 at 0x1000
```

In the following example, the command did not function because it must be entered in lowercase:

```
rommon 10 > BOOT
command "BOOT" not found
```

The following example boots the first file in the first partition of internal Flash memory of a Cisco 3600 series router:

```
> boot flash:
```

The following example boots the first image file in the first partition of the Flash memory card in slot 0 of a Cisco 3600 series router:

```
> boot slot0:
```

The following example shows the ROM monitor booting the first file in the first Flash memory partition on a Cisco 1600 series router:

```
> boot flash:
```

Related Commands

Command	Description
continue	Returns to EXEC mode from ROM monitor mode by completing the boot process.

boot bootldr

To specify the location of the boot image that ROM uses for booting, use the **boot bootldr** command in global configuration mode. To remove this boot image specification, use the **no** form of this command.

boot bootldr *file-url*

no boot bootldr

Syntax Description

<i>file-url</i>	URL of the boot image on a Flash file system.
-----------------	---

Defaults

Refer to your platform documentation for the location of the default boot image.

Command Modes

Global configuration

Command History

Release	Modification
11.0	The command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The **boot bootldr** command sets the BOOTLDR variable in the current running configuration. You must specify both the Flash file system and the filename.



Note

When you use this global configuration command, you affect only the running configuration. You must save the variable setting to your startup configuration to place the information under ROM monitor control and to have the variable function as expected. Use the **copy system:running-config nvram:startup-config** command to save the variable from your running configuration to your startup configuration.

The **no** form of the command sets the BOOTLDR variable to a null string. On the Cisco 7000 family routers, a null string causes the first image file in boot Flash memory to be used as the boot image that ROM uses for booting.

Use the **show boot** command to display the current value for the BOOTLDR variable.

Examples

In the following example, the internal Flash memory contains the boot image:

```
boot bootldr bootflash:boot-image
```

The following example specifies that the Flash memory card inserted in slot 0 contains the boot image:

```
boot bootldr slot0:boot-image
```

Related Commands

Command	Description
copy system:running-config nvram:startup-config	Copies any file from a source to a destination.
show (flash file system)	Displays the layout and contents of a Flash memory file system.
show bootvar	Displays the contents of the BOOT variable, the name of the configuration file pointed to by the CONFIG_FILE variable, the contents of the BOOTLDR variable, and the configuration register setting.

boot bootstrap

To configure the filename that is used to boot a secondary bootstrap image, use the **boot bootstrap** command in global configuration mode. To disable booting from a secondary bootstrap image, use the **no** form of this command.

boot bootstrap *file-url*

no boot bootstrap *file-url*

boot bootstrap flash [*filename*]

no boot bootstrap flash [*filename*]

boot bootstrap [tftp] *filename* [*ip-address*]

no boot bootstrap [tftp] *filename* [*ip-address*]

Syntax Description		
	<i>file-url</i>	URL of the bootstrap image.
	flash	Boots the router from Flash memory.
	<i>filename</i>	(Optional with flash) Name of the system image to boot from a network server or from Flash memory. If you omit the filename when booting from Flash memory, the router uses the first system image stored in Flash memory.
	tftp	(Optional) Boots the router from a system image stored on a TFTP server.
	<i>ip-address</i>	(Optional) IP address of the TFTP server on which the system image resides. If omitted, this value defaults to the IP broadcast address of 255.255.255.255.

Defaults No secondary bootstrap

Command Modes Global configuration

Command History	Release	Modification
	10.0	This command was introduced.
	12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines The **boot bootstrap** command causes the router to load a secondary bootstrap image from the specified URL, such as from a remote server. After the bootstrap image is loaded, the bootstrap image loads the specified system image file. See the appropriate hardware installation guide for details on the setting the configuration register and secondary bootstrap filename.

Use this command when you have attempted to load a system image but have run out of memory even after compressing the system image. Secondary bootstrap images allows you to load a larger system image through a smaller secondary image.

Examples

In the following example, the system image file named sysimage-2 will be loaded by using a secondary bootstrap image:

```
Router(config)# boot bootstrap bootflash:sysimage-2
```

boot config

To specify the device and filename of the configuration file from which the system configures itself during initialization (startup), use the **boot config** command in global configuration mode. To return to the default location for the configuration file, use the **no** form of this command.

Platforms Other than Cisco 7600 Series Router

```
boot config file-system-prefix:[directory/]filename [nvbypass]
```

```
no boot config
```

Cisco 7600 Series Router

```
boot config device:filename [nvbypass]
```

```
no boot config
```

Syntax Description		
<i>file-system-prefix:</i>		File system, followed by a colon (for example, nvr am:, flash :, slot 0:, usbflash [0-9]:, or usbtoken [0-9]:). The default is nvr am:.
<i>directory/</i>		(Optional) File system directory where the configuration file is located, followed by a forward slash (/).
<i>filename</i>		Name of the configuration file.
<i>device:</i>		Device identification, followed by a colon; see the “Usage Guidelines” section for a list of the valid values.
nvbypass		(Optional) Specifies that the distilled configuration is not written to nonvolatile random access memory (NVRAM).

Command Default The default location for the configuration file is NVRAM (**nvr**am:).

Command Modes Global configuration (config)

Command History	Release	Modification
	12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
	12.2(14)SX	Support for this command was added for the Cisco 7600 Supervisor Engine 720.
	12.2(17d)SXB	Support for this command on the cisco 7600 Supervisor Engine 2 was extended to Release 12.2(17d)SXB.
	11.0	This command was introduced.
	12.3(14)T	Support for Class B file system platforms and the following file system prefix options were added: usbflash [0-9]: and usbtoken [0-9]:.

Supported Platforms Other than Cisco 7600 Series Router

This command is available only on Class A and Class B file system platforms.

You set the CONFIG_FILE environment variable in the current running memory when you use the **boot config** command. This variable specifies the configuration file used for initialization (startup). The configuration file must be an ASCII file located in either NVRAM or flash memory.

Cisco 7600 Series Router

The valid values for the *device:* argument and colon are as follows:

- For systems that are configured with a Supervisor Engine 2, the valid values are **bootflash:**, **const_nvram:**, **flash:**, **nvrasm:**, **slot0:**, **sup-slot0:**, and **sup-bootflash:**.
- For systems that are configured with a Supervisor Engine 720, the valid values are **disk0:** and **disk1:**.

The configuration file must be an ASCII file that is located in the specified file system.

The **disk0:** and **disk1:** keywords are for Class C file systems.

The **bootflash:**, **slot0:**, and **sup-bootflash:** keywords are for Class A file systems.

For Class A flash file systems, the CONFIG_FILE environment variable specifies the file system and filename of the configuration file to use for initialization (startup). You set the CONFIG_FILE environment variable in the current running memory when you use the **boot config** command. This variable specifies the configuration file used for initialization (startup).

All Supported Platforms

When you use the **boot config** command, you affect only the running configuration. You must save the environment variable setting to your startup configuration to place the information under ROM monitor control and to have the environment variable function as expected. Use the **copy system:running-config nvram:startup-config** command to save the environment variable from your running configuration to your startup configuration.

The software displays an error message and does not update the CONFIG_FILE environment variable in the following situations:

- You specify **nvrasm:** as the file system, and it contains only a distilled version of the configuration. (A distilled configuration is one that does not contain access lists.)
- You specify a configuration file in the *filename* argument that does not exist or is not valid.

The router uses the NVRAM configuration during initialization when the CONFIG_FILE environment variable does not exist or when it is null (such as at first-time startup). If the software detects a problem with NVRAM or the configuration it contains, the device enters setup mode.

When you use the **no** form of this command, the router returns to using the default NVRAM configuration file as the startup configuration.

You can display the contents of the BOOT, BOOTLDR, and the CONFIG_FILE environment variables using the **show bootvar** command. This command displays the settings for these variables as they exist in the startup configuration and in the running configuration if a running configuration setting differs from a startup configuration setting.

When the **boot config** command is used, the distilled configuration is written into NVRAM and the system configuration is written into the file specified by the **boot config** command. If the distilled configuration exceeds the size of NVRAM, the system configuration gets truncated. Use the **nvbypass** keyword to prevent the system configuration from being truncated when the distilled configuration is larger than the size of NVRAM.

Examples

The following example shows how to set the configuration file that is located in internal flash memory to configure itself during initialization. The third line copies the specification to the startup configuration, ensuring that this specification will take effect upon the next reload.

```
Router(config)# boot config flash:router-config
Router(config)# end
Router# copy system:running-config nvram:startup-config
```

The following example instructs a Cisco 7500 series router to use the configuration file named router-config located on the flash memory card inserted in the second Personal Computer Memory Card Industry Association (PCMCIA) slot of the Route Switch Processor (RSP) card during initialization. The third line copies the specification to the startup configuration, ensuring that this specification will take effect upon the next reload.

```
Router (config)# boot config slot1:router-config
Router (config)# end
Router# copy system:running-config nvram:startup-config
```

Related Commands

Command	Description
copy system:running-config nvram:startup-config	Saves the environment variable from the running configuration to the startup configuration.
show bootvar	Displays the contents of the BOOT environment variable, the name of the configuration file pointed to by the CONFIG_FILE environment variable, the contents of the BOOTLDR environment variable, and the configuration register setting.

boot host

To specify the host-specific configuration file to be used at the next system startup, use the **boot host** command in global configuration mode. To restore the host configuration filename to the default, use the **no** form of this command.

boot host *remote-url*

no boot host *remote-url*

Syntax Description

<i>remote-url</i>	Location of the configuration file. Use the following syntax: <ul style="list-style-type: none"> • ftp:[[[//[<i>username</i>[:<i>password</i>]]@]<i>location</i>]/<i>directory</i>]/<i>filename</i>] • rcp:[[[//[<i>username</i>@]<i>location</i>]/<i>directory</i>]/<i>filename</i>] • tftp:[[[//<i>location</i>]/<i>directory</i>]/<i>filename</i>]
-------------------	---

Defaults

If you do not specify a *filename* using this command, the router uses its configured host name to request a configuration file from a remote server. To form the configuration filename, the router converts its name to all lowercase letters, removes all domain information, and appends *-confg* or *-config*.

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

This command instructs the system to “Boot using host-specific configuration file *x*,” where *x* is the filename specified in the *remote-url* argument. In other words, this command specifies the remote location and filename of the host-specific configuration file to be used at the next system startup, as well as the protocol to be used to obtain the file.

Before using the **boot host** command, use the **service config** global configuration command to enable the loading of the specified configuration file at reboot time. Without this command, the router ignores the **boot host** command and uses the configuration information in NVRAM. If the configuration information in NVRAM is invalid or missing, the **service config** command is enabled automatically.

The network server will attempt to load two configuration files from remote hosts. The first is the network configuration file containing commands that apply to all network servers on a network. Use the **boot network** command to identify the network configuration file. The second is the host configuration file containing commands that apply to one network server in particular. Use the **boot host** command to identify the host configuration file.

**Note**

Usually, the **service config** command is used in conjunction with the **boot host** or **boot network** command. You must enter the **service config** command to enable the router to automatically configure the system from the file specified by the **boot host** or **boot network** command.

With IOS software versions 12.3(2)T , 12.3(1)B, and later, you no longer have to specify the **service config** command for the **boot host** or **boot network** command to be active.

If you specify both the **no service config** command and the **boot host** command, the router attempts to find the specified host configuration file. The **service config** command can also be used without the **boot host** or **boot network** command. If you do not specify host or network configuration filenames, the router uses the default configuration files. The default network configuration file is network-config. The default host configuration file is host-config, where host is the hostname of the router. If the Cisco IOS software cannot resolve its hostname, the default host configuration file is router-config.

Loading a Configuration File Using rcp

The rcp software requires that a client send the remote username on each rcp request to the network server. If the server has a directory structure (such as UNIX systems), the rcp implementation searches for the configuration files starting in the directory associated with the remote username.

When you load a configuration file from a server using rcp, the Cisco IOS software sends the first valid username in the following list:

1. The username specified in the file-URL, if a username is specified.
2. The username set by the **ip rcmd remote-username** command, if the command is configured.
3. The router host name.

**Note**

An account for the username must be defined on the destination server. If the network administrator of the destination server did not establish an account for the username, this command will not execute successfully.

Loading a Configuration File Using FTP

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. The username and password must be associated with an account on the FTP server. If the server has a directory structure, the configuration file or image copied from the directory is associated with the username on the server. Refer to the documentation for your FTP server for more details.

When you load a configuration file from a server using FTP, the Cisco IOS software sends the first valid username in the following list:

1. The username specified in the **boot host** command, if a username is specified.
2. The username set by the **ip ftp username** command, if the command is configured.
3. Anonymous.

The router sends the first valid password in the following list:

1. The password specified in the **boot host** command, if a password is specified.
2. The password set by the **ip ftp password** command, if the command is configured.

3. The router forms a password *username@routername.domain*. The variable *username* is the username associated with the current session, *routername* is the configured host name, and *domain* is the domain of the router.

Examples

The following example sets the host filename to wilma-config at address 192.168.7.19:

```
Router(config)# boot host tftp://192.168.7.19/usr/local/tftpd/wilma-config  
Router(config)# service config
```

Related Commands

Command	Description
boot network	Specifies the remote location and filename of the network configuration file to be used at the next system boot (startup).
service config	Enables autoloading of configuration files from a network server.

boot network

To change the default name of the network configuration file from which to load configuration commands, use the **boot network** command in global configuration mode. To restore the network configuration filename to the default, use the **no** form of this command.

```
boot network remote-url
```

```
no boot network remote-url
```

Syntax Description

remote-url

Location of the configuration file. Use the following syntax:

- **ftp:**[[[//*username[:password]@location/directory/*filename]]]
- **rep:**[[[//*username@location/directory/*filename]]]
- **tftp:**[[[//*location/directory/*filename]]]

Defaults

The default *filename* is network-config.

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

This command instructs the system to “Boot using network configuration file *x*,” where *x* is the filename specified in the *remote-url* argument. This command specifies the remote location and filename of the network configuration file to be used at the next system startup, as well as the protocol to be used to obtain the file.

When booting from a network server, routers ignore routing information, static IP routes, and bridging information. As a result, intermediate routers are responsible for handling FTP, rcp, or TFTP requests. Before booting from a network server, verify that a server is available by using the **ping** command.

Use the **service config** command to enable the loading of the specified configuration file at reboot time. Without this command, the router ignores the **boot network** command and uses the configuration information in NVRAM. If the configuration information in NVRAM is invalid or missing, the **service config** command is enabled automatically.

The network server will attempt to load two configuration files from remote hosts. The first is the network configuration file containing commands that apply to all network servers on a network. Use the **boot network** command to identify the network configuration file. The second is the host configuration file containing commands that apply to one network server in particular. Use the **boot host** command to identify the host configuration file.

**Note**

Usually, the **service config** command is used in conjunction with the **boot host** or **boot network** command. You must enter the **service config** command to enable the router to automatically configure the system from the file specified by the **boot host** or **boot network** command.

With IOS software versions 12.3(2)T , 12.3(1)B, and later, you no longer have to specify the **service config** command for the **boot host** or **boot network** command to be active.

If you specify both the **no service config** command and the **boot host** command, the router attempts to find the specified host configuration file. The **service config** command can also be used without the **boot host** or **boot network** command. If you do not specify host or network configuration filenames, the router uses the default configuration files. The default network configuration file is network-config. The default host configuration file is host-config, where host is the hostname of the router. If the Cisco IOS software cannot resolve its hostname, the default host configuration file is router-config.

Loading a Configuration File Using rcp

The rcp software requires that a client send the remote username on each rcp request to the network server. If the server has a directory structure (such as UNIX systems), the rcp implementation searches for the configuration files starting in the directory associated with the remote username.

When you load a configuration file from a server using rcp, the Cisco IOS software sends the first valid username in the following list:

1. The username specified in the file-URL, if a username is specified.
2. The username set by the **ip rcmd remote-username** command, if the command is configured.
3. The router host name.

**Note**

An account for the username must be defined on the destination server. If the network administrator of the destination server did not establish an account for the username, this command will not execute successfully.

Loading a Configuration File Using FTP

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. The username and password must be associated with an account on the FTP server. If the server has a directory structure, the configuration file or image copied from the directory associated with the username on the server. Refer to the documentation for your FTP server for more details.

When you load a configuration file from a server using FTP, the Cisco IOS software sends the first valid username in the following list:

1. The username specified in the **boot network** command, if a username is specified.
2. The username set by the **ip ftp username** command, if the command is configured.
3. Anonymous.

The router sends the first valid password in the following list:

1. The password specified in the **boot network** command, if a password is specified.
2. The password set by the **ip ftp password** command, if the command is configured.

- The router forms a password *username@routername.domain*. The variable *username* is the username associated with the current session, *routername* is the configured host name, and *domain* is the domain of the router.

Examples

The following example changes the network configuration filename to `bridge_9.1` and uses the default broadcast address:

```
Router(config)# boot network tftp:bridge_9.1
Router(config)# service config
```

The following example changes the network configuration filename to `bridge_9.1`, specifies that `rcp` is to be used as the transport mechanism, and gives `172.16.1.111` as the IP address of the server on which the network configuration file resides:

```
Router(config)# service config
Router(config)# boot network rcp://172.16.1.111/bridge_9.1
```

Related Commands

Command	Description
<code>boot host</code>	Specifies the remote location and filename of the host-specific configuration file to be used at the next system boot (startup).
<code>service config</code>	Enables autoloading of configuration files from a remote host.

boot system

To specify the system image that the router loads at startup, use one of the following **boot system** command in global configuration mode. To remove the startup system image specification, use the **no** form of this command.

Loading System Image from a URL or a TFTP File

```
boot system {file-url | filename}
```

```
no boot system {file-url | filename}
```

Booting from a System Image in Internal Flash

```
boot system flash [flash-fs:] [partition-number:] [filename]
```

```
no boot system flash [flash-fs:] [partition-number:] [filename]
```

Booting from a MOP Server

```
boot system mop filename [mac-address] [interface]
```

```
no boot system mop filename [mac-address] [interface]
```

Booting from ROM

```
boot system rom
```

```
no boot system rom
```

Booting a System Image from a Network, TFTP, or FTP Server

```
boot system {rcp | tftp | ftp} filename [ip-address]
```

```
no boot system {rcp | tftp | ftp} filename [ip-address]
```

Syntax Description	
<i>file-url</i>	The URL of the system image to load at system startup.
<i>filename</i>	The TFTP filename of the system image to load at system startup.
flash	<p>On all platforms except the Cisco 1600 series, Cisco 3600 series, and Cisco 7000 family routers, this keyword boots the router from internal flash memory. If you omit all arguments that follow this keyword, the system searches internal Flash for the first bootable image.</p> <p>On the Cisco 1600 series, Cisco 3600 series, and Cisco 7000 family routers, this keyword boots the router from the flash system specified by the <i>flash-fs:</i> argument. On the Cisco 1600 series and Cisco 3600 series routers, if you omit all optional arguments, the router searches internal flash memory for the first bootable image. On the Cisco 7000 family routers, when you omit all arguments that follow this keyword, the system searches the Personal Computer Memory Card Industry Association (PCMCIA) slot 0 for the first bootable image.</p>

<i>flash-fs:</i>	<p>(Optional) Flash file system containing the system image to load at startup. The colon is required. Valid file systems are as follows:</p> <ul style="list-style-type: none"> • flash:—Internal flash memory on the Cisco 1600 series and Cisco 3600 series routers. For the Cisco 1600 series and Cisco 3600 series routers, this file system is the default if you do not specify a file system. This is the only valid file system for the Cisco 1600 series. • bootflash:—Internal flash memory in the Cisco 7000 family. • slot0:—First PCMCIA slot on the Cisco 3600 series and Cisco 7000 family routers. For the Cisco 7000 family routers, this file system is the default if you do not specify a file system. • slot1:—Flash memory card in the second PCMCIA slot on the Cisco 3600 series and Cisco 7000 family routers. <p>On the Cisco 2600 series routers, a file system should be specified. Otherwise, the router may attempt to load the Cisco IOS software twice with unexpected results.</p>
<i>partition-number:</i>	<p>(Optional) Number of the flash memory partition that contains the system image to boot, specified by the optional <i>filename</i> argument. If you do not specify a filename, the router loads the first valid file in the specified partition of flash memory. This argument is valid only on routers that can be partitioned.</p>
<i>filename</i>	<p>(Optional when used with the boot system flash command) Name of the system image to load at startup. This argument is case sensitive. If you do not specify a value for the <i>filename</i> argument, the router loads the first valid file in the following:</p> <ul style="list-style-type: none"> • The specified flash file system • The specified partition of flash memory • The default flash file system if you also omitted the <i>flash-fs:</i> argument
mop	<p>Boots the router from a system image stored on a DECNET Maintenance Operations Protocol (MOP) server. Do not use this keyword with the Cisco 3600 series or Cisco 7000 family routers.</p>
<i>mac-address</i>	<p>(Optional) MAC address of the MOP server containing the specified system image file. If you do not include the MAC address argument, the router sends a broadcast message to all MOP boot servers. The first MOP server to indicate that it has the specified file is the server from which the router gets the boot image.</p>
<i>interface</i>	<p>(Optional) Interface the router uses to send out MOP requests to the MOP server. The interface options are async, dialer, ethernet, serial, and tunnel. If you do not specify the <i>interface</i> argument, the router sends a request out on all interfaces that have MOP enabled. The interface that receives the first response is the interface the router uses to load the software.</p>
rom	<p>Boots the router from ROM. Do not use this keyword with the Cisco 3600 series or the Cisco 7000 family routers.</p>
rcp	<p>Boots the router from a system image stored on a network server using rcp.</p>
tftp	<p>Boots the router from a system image stored on a TFTP server.</p>

ftp	Boots the router from a system image stored on an FTP server.
<i>ip-address</i>	(Optional) IP address of the server containing the system image file. If omitted, this value defaults to the IP broadcast address of 255.255.255.255.

Command Default

If you configure the router to boot from a network server but do not specify a system image file with the **boot system** command, the router uses the configuration register settings to determine the default system image filename. The router forms the default boot filename by starting with the word *cisco* and then appending the octal equivalent of the boot field number in the configuration register, followed by a hyphen (-) and the processor type name (*cisconn-cpu*). Refer to the appropriate hardware installation guide for details on the configuration register and default filename. See also the **config-register** or **confreg** command.

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(14)SX	Support for this command was added for the Supervisor Engine 720.
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was extended to Release 12.2(17d)SXB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(31)SB2	This command was integrated into Cisco IOS Release 12.2(31)SB.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Usage Guidelines

For this command to work, the **config-register** command must be set properly.

Enter several **boot system** commands to provide a fail-safe method for booting your router. The router stores and executes the **boot system** commands in the order in which you enter them in the configuration file. If you enter multiple boot commands of the same type—for example, if you enter two commands that instruct the router to boot from different network servers—then the router tries them in the order in which they appear in the configuration file. If a **boot system** command entry in the list specifies an invalid device, the router omits that entry. Use the **boot system rom** command to specify use of the ROM system image as a backup to other **boot system** commands in the configuration.

For some platforms, the boot image must be loaded before the system image is loaded. However, on many platforms, the boot image is loaded only if the router is booting from a network server or if the flash file system is not specified. If the file system is specified, the router will boot faster because it need not load the boot image first.

This section contains the following topics:

- Changing the List of Boot System Commands
- Booting Compressed Images
- Understanding rep
- Understanding TFTP
- Understanding FTP

- Stopping Booting and Entering ROM Monitor Mode
- Cisco 1600 Series, Cisco 3600 Series, Cisco 7000 Family, and Cisco 7600 Series Router Notes

Changing the List of Boot System Commands

To remove a single entry from the bootable image list, use the **no** form of the command with an argument. For example, to remove the entry that specifies a bootable image on a flash memory card inserted in the second slot, use the **no boot system flash slot1:[filename]** command. All other entries in the list remain.

To eliminate all entries in the bootable image list, use the **no boot system** command. At this point, you can redefine the list of bootable images using the previous **boot system** commands. Remember to save your changes to your startup configuration by issuing the **copy system:running-config nvram:startup-config** command.

Each time you write a new software image to flash memory, you must delete the existing filename in the configuration file with the **no boot system flash filename** command. Then add a new line in the configuration file with the **boot system flash filename** command.



Note

If you want to rearrange the order of the entries in the configuration file, you must first issue the **no boot system** command and then redefine the list.

Booting Compressed Images

You can boot the router from a compressed image on a network server. When a network server boots software, both the image being booted and the running image must be able to fit into memory. Use compressed images to ensure that enough memory is available to boot the router. You can compress a software image on any UNIX platform using the **compress** command. Refer to your UNIX platform's documentation for the exact usage of the **compress** command. (You can also uncompress data with the UNIX **uncompress** command.)

Understanding rcp

The rcp requires that a client send the remote username in an rcp request to a server. When the router executes the **boot system rcp** command, the Cisco IOS software sends the hostname as both the remote and local usernames by default. Before the rcp can execute properly, an account must be defined on the network server for the remote username configured on the router.

If the server has a directory structure, the rcp software searches for the system image to boot from the remote server relative to the directory of the remote username.

By default, the router software sends the hostname as the remote username. You can override the default remote username by using the **ip rcmd remote-username** command. For example, if the system image resides in the home directory of a user on the server, you can specify that user's name as the remote username.

Understanding TFTP

You need a TFTP server running to retrieve the router image from the host.

Understanding FTP

You need an FTP server running to retrieve the router image from the host. You also need an account on the server or anonymous file access to the server.

Stopping Booting and Entering ROM Monitor Mode

During the first 60 seconds of startup, you can force the router to stop booting by pressing the Break key. The router will enter ROM monitor mode, where you can change the configuration register value or boot the router manually.

Cisco 1600 Series, Cisco 3600 Series, Cisco 7000 Family, and Cisco 7600 Series Router Notes

For the Cisco 3600 series and Cisco 7000 family, the **boot system** command modifies the BOOT variable in the running configuration. The BOOT variable specifies a list of bootable images on various devices.



Note

When you use the **boot system** command on the Cisco 1600 series, Cisco 3600 series, Cisco 7000 family, and Cisco 7600 series, you affect only the running configuration. You must save the BOOT variable settings to your startup configuration to place the information under ROM monitor control and to have the variable function as expected. Use the **copy system:running-config nvram:startup-config** privileged EXEC command to save the variable from your running configuration to your startup configuration.

To display the contents of the BOOT variable, use the **show bootvar** EXEC command.

Examples

The following example illustrates a configuration that specifies two possible internetwork locations for a system image, with the ROM software being used as a backup:

```
Router(config)# boot system tftp://192.168.7.24/cs3-rx.90-1
Router(config)# boot system tftp://192.168.7.19/cs3-rx.83-2
Router(config)# boot system rom
```

The following example boots the system boot relocatable image file named igs-bpx-1 from partition 2 of the flash device:

```
Router(config)# boot system flash:2:igs-bpx-1
```

The following example instructs the router to boot from an image located on the flash memory card inserted in slot 0:

```
Router(config)# boot system slot0:new-config
```

The following example specifies the file named new-ios-image as the system image for a Cisco 3600 series router to load at startup. This file is located in the fourth partition of the flash memory card in slot 0.

```
Router(config)# boot system slot0:4:dirt/images/new-ios-image
```

This example boots from the image file named c1600-y-1 in partition 2 of flash memory of a Cisco 1600 series router:

```
Router(config)# boot system flash:2:c1600-y-1
```

Related Commands

Command	Description
boot	Boots the router manually.
config-register	Changes the configuration register settings.

Command	Description
confreg	Changes the configuration register settings while in ROM monitor mode.
copy	Copies any file from a source to a destination.
copy system:running-config nvram:startup-config	Copies the running configuration to the startup configuration.
ip rcmd remote username	Configures the remote username to be used when requesting a remote copy using rcp.
show bootvar	Displays the contents of the BOOT variable, the name of the configuration file pointed to by the CONFIG_FILE variable, the contents of the BOOTLDR variable, and the configuration register setting.

boot-end-marker

The **boot-start-marker** and **boot-end-marker** flags, which can be seen in Cisco IOS software configuration files, are not CLI commands. These markers are written to configuration files automatically to flag the beginning and end of the boot commands (boot statements). By flagging boot statements, these markers allow the router to more reliably load Cisco IOS images during bootup.

A boot statement is one or more lines in a configuration file that tells the router which software image to load after a powercycling (reboot). The boot-start-marker and boot-end-marker flags will appear around any boot commands, including:

- **boot bootstrap**
- **boot config**
- **boot host**
- **boot network**
- **boot system**

Note, however, that these markers will always appear in the output of the **show running-config** or **more system:running-config** commands, regardless of whether any actual boot commands have been entered. This means that no boot commands will appear between the markers if no boot commands have been entered, or if all boot commands have been removed from the configuration, as shown in the “Examples” section.

The **boot-start-marker** and **boot-end-marker** flags cannot be removed or modified using the CLI. These markers are written to the startup configuration file whenever a **copy running-config startup-config** command is issued.

These flags were also introduced to circumvent errors in the configuration file, such as a leading space before a boot command (such as those sometimes introduced by manually edited configuration files), or the use of text strings that include the word “boot” in banners or other user-specified text.

If the “boot start-marker” flag is not found in the configuration file, the system will use the traditional method to identify the boot commands. However, if you are manually creating configuration files, or copying from older Cisco IOS software releases, the addition of these markers is recommended.

Command History

Release	Modification
12.3(3), 12.3(4)T, 12.0(26)S, 12.0(27)SV, 12.3(3)B,	The boot-start-marker and boot-end-marker flags were introduced.

Examples

In the following example, a **boot** command is entered, and the boot-start-marker and boot-end-marker flags are shown in the context of the startup configuration file:

```
Router# configure terminal
Enter configuration commands, one per line. End with the end command.

Router(config)# boot system slot0:
Router(config)# end
Router# copy running-config startup-config
Router# show startup-config

Using 1398 out of 129016 bytes
!
```

```

version 12.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname C3660-2
!
boot-start-marker
boot system slot0:
boot-end-marker
!
logging count
.
.
.

```

In the following example, the boot-start-marker and boot-end-marker flags appear in the configuration file even though no **boot** commands have been entered:

```

Router# show running-configuration

Current configuration :3055 bytes
!
! No configuration change since last restart
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
.
.
.

```

Related Commands

Command	Description
boot bootstrap	Specifies the filename and location of a secondary bootstrap image (to be used if a valid software image cannot be loaded).
boot config	Specifies the device and filename of the configuration file from which the router boots during startup (for Class A filesystems).
boot host	Specifies a remote host location for the host-specific (router-specific) configuration file to be used at the next system startup.
boot network	Specifies a remote location for the network (network-wide) configuration file to be used at the next system startup.
boot system	Specifies the system software image that the router loads at startup.

boot-start-marker

The **boot-start-marker** and **boot-end-marker** flags, which can be seen in Cisco IOS software configuration files, are not CLI commands. These markers are written to configuration files automatically to flag the beginning and end of the boot commands (boot statements). By flagging boot statements, these markers allow the router to more reliably load Cisco IOS images during bootup.

A boot statement is one or more lines in a configuration file that tells the router which software image to load after a powercycling (reboot). The boot-start-marker and boot-end-marker flags will appear around any boot commands, including:

- **boot bootstrap**
- **boot config**
- **boot host**
- **boot network**
- **boot system**

Note, however, that these markers will always appear in the output of the **show running-config** or **more system:running-config** commands, regardless of whether any actual boot commands have been entered. This means that no boot commands will appear between the markers if no boot commands have been entered, or if all boot commands have been removed from the configuration, as shown in the “Examples” section.

The **boot-start-marker** and **boot-end-marker** flags cannot be removed or modified using the CLI. These markers are written to the startup configuration file whenever a **copy running-config startup-config** command is issued.

These flags were also introduced to circumvent errors in the configuration file, such as a leading space before a boot command (such as those sometimes introduced by manually edited configuration files), or the use of text strings that include the word “boot” in banners or other user-specified text.

If the “boot start-marker” flag is not found in the configuration file, the system will use the traditional method to identify the boot commands. However, if you are manually creating configuration files, or copying from older Cisco IOS software releases, the addition of these markers is recommended.

Command History

Release	Modification
12.3(3), 12.3(4)T, 12.0(26)S, 12.0(27)SV, 12.3(3)B	The boot-start-marker and boot-end-marker flags were introduced.

Examples

In the following example, a **boot** command is entered, and the boot-start-marker and boot-end-marker flags are shown in the context of the startup configuration file:

```
Router# configure terminal
Enter configuration commands, one per line. End with the end command.

Router(config)# boot system slot0:
Router(config)# end
Router# copy running-config startup-config
Router# show startup-config

Using 1398 out of 129016 bytes
!
```

```

version 12.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname C3660-2
!
boot-start-marker
boot system slot0:
boot-end-marker
!
logging count
.
.
.

```

In the following example, the boot-start-marker and boot-end-marker flags appear in the configuration file even though no **boot** commands have been entered:

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

```

Current configuration :3055 bytes
!
! No configuration change since last restart
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
.
.
.

```

Related Commands

Command	Description
boot bootstrap	Specifies the filename and location of a secondary bootstrap image (to be used if a valid software image cannot be loaded).
boot config	Specifies the device and filename of the configuration file from which the router boots during startup (for Class A filesystems).
boot host	Specifies a remote host location for the host-specific (router-specific) configuration file to be used at the next system startup.
boot network	Specifies a remote location for the network (network-wide) configuration file to be used at the next system startup.
boot system	Specifies the system software image that the router loads at startup.