System Image and Configuration File Load Commands

This chapter provides detailed descriptions of the commands used to load and copy system images and configuration files. System images contain the system software. Configuration files contain commands entered to customize the function of the communication server.

For communication server configuration information and examples, refer to the "Loading System Images and Configuration Files" chapter in the *Access and Communication Servers Configuration Guide*.

async-bootp

Use the **async-bootp** command to enable support for extended BOOTP requests as defined in RFC 1084 when the communication server is configured for SLIP. Use the **no** form of this command to restore the default.

> **async-bootp** *tag* [*:hostname*] *data*
> **no async-bootp**

Syntax Description

tag
Item being requested; expressed as filename, integer, or IP dotted-decimal address. See Table 3-1 for possible values.

:hostname
(Optional) This entry applies only to the host specified. The argument *:hostname* accepts both an IP address and a logical host name.

data
List of IP addresses entered in dotted-decimal notation or as logical host names, a number, or a quoted string.

Table 3-1   Async-BOOTP Tag Keywords

Keyword	Description
bootfile	Specifies use of a server boot file from which to download the boot program. Use the optional *:hostname* and *data* arguments to specify the filename.
subnet-mask *mask*	Dotted-decimal address specifying the network and local subnetwork mask (as defined by RFC 950).
time-offset *offset*	Signed 32-bit integer specifying the time offset of the local subnetwork in seconds from Universal Coordinated Time (UTC).
gateway *address*	Dotted-decimal address specifying the IP addresses of gateways for this subnetwork. A preferred gateway should be listed first.
time-server *address*	Dotted-decimal address specifying the IP address of time servers (as defined by RFC 868).
ien116-server *address*	Dotted-decimal address specifying the IP address of name servers (as defined by IEN 116).
dns-server *address*	Dotted-decimal address specifying the IP address of the Domain Name Server (DNS) (as defined by RFC 1034).
log-server *address*	Dotted-decimal address specifying the IP address of an MIT-LCS UDP log server.
quote-server *address*	Dotted-decimal address specifying the IP address of Quote of the Day servers (as defined in RFC 865).
lpr-server *address*	Dotted-decimal address specifying the IP address of Berkeley UNIX Version 4 BSD servers.
impress-server *address*	Dotted-decimal address specifying the IP address of Impress network image servers.
rlp-server *address*	Dotted-decimal address specifying the IP address of Resource Location Protocol (RLP) servers (as defined in RFC 887).
hostname *name*	The name of the client, which might or might not be domain qualified, depending upon the site.
bootfile-size *value*	A two-octet value specifying the number of 512-octet (byte) blocks in the default boot file.

Default

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

Command Mode

Global configuration

Usage Guidelines

Use the EXEC command **show async bootp** 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 128.128.1.1 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 :128.128.1.1 "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 128.128.1.1
```

Related Command

show async bootp

b

To boot the communication server manually, use the **b** ROM monitor command.

> **b**
> **b** *filename* [*ip-address*]
> **b flash** [*filename*]
> **b flash** [*device***:**][*partition-number***:**][*filename*]

Syntax Description

filename	Name of the system image from which you want to netboot. The filename is case sensitive.
ip-address	(Optional) IP address of the network server on which the system image resides. If omitted, this value defaults to the IP broadcast address of 255.255.255.255.
flash *filename*	(Optional) Boots the communication server from Flash memory with the optional filename of the image you want loaded. The filename is case sensitive. Without a filename, the first valid file in Flash memory will be loaded.
*device***:**	(Optional) Valid value is **flash**.
[*partition-number***:**]	(Optional) Boots the communication server from Flash memory with the optional filename of the image you want loaded from the specified Flash partition. If you do not specify a filename, the first valid file in the specified partition of Flash memory is loaded.
filename	(Optional) Boots the communication server from Flash memory with the filename of the image you want loaded from the specified Flash partition, if a partition is specified. If a partition is not specified, the system boots with the filename from the first partition. The filename is case sensitive. If you do not specify a filename, the first valid file in the specified partition of Flash memory is loaded.

Default

If you enter the **b** command and press Return, the communication server boots from ROM by default.

If you enter the **b flash** command without specifying a filename, the first valid file in Flash memory is loaded.

For other defaults, see the "Syntax Description" section.

Command Mode

ROM monitor

Usage Guidelines

Use this command only when your communication server cannot find the configuration information needed in nonvolatile memory. To get to the ROM monitor prompt (>), enter the **reload** EXEC command, and then press the **Break** key during the first 60 seconds of startup.

Examples

In the following example, the communication server is manually booted from ROM:

```
> b
F3:
(ROM Monitor copyrights)
```

In the following example, the file *cstest* is netbooted from IP address 131.108.15.112:

```
> b cstest 131.108.15.112
F3:
(ROM Monitor copyrights)
```

The following example shows the **b flash** command without the *filename* argument. The first valid file in Flash memory is loaded.

```
> b flash
F3: 1858656+45204+166896 at 0x1000

Booting gs7-k from flash memory RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR [OK - 1903912/13765276 bytes]
F3: 1858676+45204+166896 at 0x1000

(ROM Monitor copyrights)
```

In the following example, the **b flash** command is used with the filename *gs7-k*. That is the file that will be loaded.

```
> b flash gs7-k
F3: 1858656+45204+166896 at 0x1000

Booting gs7-k from flash memory RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
RRRRRRRRRRRRRR [OK - 1903912/13765276 bytes]
F3: 1858676+45204+166896 at 0x1000

(ROM Monitor copyrights)
```

In the following example, the **b flash flash** command boots the relocatable image file *igs-bpx-l* from partition 2 in Flash memory.

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

(ROM Monitor copyrights)
```

The following example shows the **b flash** command without the *filename* argument.The first valid file in Flash memory is loaded.

```
> b flash
Booting igs-bfpx.102.1 from Flash address space
F3: 3911536+96836+319604 at 0x3000060

                Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

                cisco Systems, Inc.
                1525 O'Brien Drive
                Menlo Park, California 94025


3000 Software (IGS-BFPX), Version 10.2
Copyright (c) 1986-1994 by cisco Systems, Inc.
Compiled Tue 05-Jul-94 16:14

% System running from device (System flash) being initialized.
   Setting System flash access to read-only.
SNMP Research SNMP Agent Resident Module Version 12.2.0.0
Copyright 1989, 1990, 1991, 1992, 1993, 1994 SNMP Research, Inc.
cisco 2500 (68030) processor (revision A) with 1024K/1024K bytes of memory.
Processor board serial number 01244583
X.25 software, Version 2.0, NET2, BFE and GOSIP compliant.
Bridging software.
SuperLAT software (copyright 1990 by Meridian Technology Corp).
Authorized for Enterprise software set.  (0x0)
1 Ethernet/IEEE 802.3 interface.
2 Serial network interfaces.
16 terminal lines.
32K bytes of non-volatile configuration memory.
4096K bytes of processor board System flash. (Read only mode)

Press RETURN to get started!
```

In the following example, the **boot flash** command is used with the image filename *igs-bfpx.102.1*. That is the file that will be loaded.

```
> b flash igs-bfpx.102.1
Booting igs-bfpx.102.1 from Flash address space
F3: 3911536+96836+319604 at 0x3000060


              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

              cisco Systems, Inc.
              1525 O'Brien Drive
              Menlo Park, California 94025


3000 Software (IGS-BFPX), Version 10.2
Copyright (c) 1986-1994 by cisco Systems, Inc.
Compiled Tue 05-Jul-94 16:14


% System running from device (System flash) being initialized.
   Setting System flash access to read-only.
SNMP Research SNMP Agent Resident Module Version 12.2.0.0
Copyright 1989, 1990, 1991, 1992, 1993, 1994 SNMP Research, Inc.
cisco 2500 (68030) processor (revision A) with 1024K/1024K bytes of memory.
Processor board serial number 01244583
X.25 software, Version 2.0, NET2, BFE and GOSIP compliant.
Bridging software.
SuperLAT software (copyright 1990 by Meridian Technology Corp).
Authorized for Enterprise software set.  (0x0)
1 Ethernet/IEEE 802.3 interface.
2 Serial network interfaces.
16 terminal lines.
32K bytes of non-volatile configuration memory.
4096K bytes of processor board System flash. (Read only mode)

Press RETURN to get started!
```

boot bootstrap

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

boot bootstrap flash [*filename*]
no boot bootstrap flash [*filename*]

boot bootstrap mop *filename* [*mac-address*] [*interface*]
no boot bootstrap mop *filename* [*mac-address*] [*interface*]

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

Syntax Description

flash	Indicates that the communication server will be booted from Flash memory.
mop	Indicates that the communication server will be netbooted from a system image stored on a Digital MOP server.
tftp	(Optional) Indicates that the communication server will be netbooted from a system image stored on a TFTP server.
filename	(Optional with **flash**) Name of the system image from which you want to netboot. If you omit the filename when booting from Flash, the communication server uses the first system image stored in Flash memory.
ip-address	(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.
mac-address	(Optional) MAC address of the MOP server on which the file resides. If the MAC address argument is not included, a broadcast message is sent to all MOP boot servers. The first MOP server to indicate that it has the file will be the server from which the communication server gets the boot image.
interface	(Optional) Interface out which the communication server should send MOP requests to reach the MOP server. The interface options are **async**, **dialer**, **Ethernet**, **loopback**, **null**, **serial**, and **tunnel**. If the *interface* argument is not specified, a request will be sent on all interfaces that have MOP enabled, and the interface from which the first response is received will be used to load the software.

Default

No secondary bootstrap

Command Mode

Global configuration

Usage Guidelines

The **boot bootstrap** command, in conjunction with setting bit 9 on the configuration register of a communication server, causes the communication server to load a secondary bootstrap image over the network. The secondary bootstrap image then loads the specified system image file. The name of the secondary bootstrap file is *boot-csc3* or *boot-csc4*, depending on the communication server model. See the appropriate hardware installation guide for details on 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 allows you to load a larger system image through a smaller secondary image.

Example

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

```
boot bootstrap sysimage-2
```

boot buffersize

To modify the buffer size used to load configuration files, use the **boot buffersize** global configuration command. Use the **no** form of the command to return to the default setting.

boot buffersize *bytes*
no boot buffersize

Syntax Description

bytes	Specifies the size of the buffer to be used.

Default

Buffer size of the nonvolatile memory

Command Mode

Global configuration

Usage Guidelines

Normally, the communication server uses a buffer the size of the system nonvolatile memory to hold configuration commands read from the network. You can increase this size if you have a very complex configuration. There is no minimum or maximum size that can be specified.

Example

The following example sets the buffer size to 64000:

```
boot buffersize 64000
```

boot host

To change the default name of the host configuration filename from which you want to load configuration commands, use the **boot host** global configuration command. Use the **no** form of the command to restore the host configuration filename to the default.

boot host mop *filename* [*mac-address*] [*interface*]
no boot host mop *filename* [*mac-address*] [*interface*]

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

Syntax Description

mop	Indicates that the communication server will be configured from a configuration file stored on a Digital MOP server.
tftp	(Optional) Indicates that the communication server will be configured from a configuration file stored on a TFTP server.
filename	Name of the file from which you want to load configuration commands.
ip-address	(Optional) IP address of the TFTP server on which the file resides. If omitted, this value defaults to the IP broadcast address of 255.255.255.255.
mac-address	(Optional) MAC address of the MOP server on which the file resides. If the MAC address argument is not included, a broadcast message is sent to all MOP boot servers. The first MOP server to indicate that it has the file will be the server from which the communication server gets the boot image.
interface	(Optional) Interface out which the communication server should send MOP requests to reach the MOP server. The interface options are **async**, **dialer**, **ethernet**, **serial**, and **tunnel**. If the *interface* argument is not specified, a request will be sent on all interfaces that have MOP enabled, and the interface from which the first response is received will be used to load the software.

Default

The communication server uses its host name to form a host configuration filename. To form this name, the communication server converts its name to all lowercase letters, removes all domain information, and appends *-confg*.

Command Mode

Global configuration

Usage Guidelines

Use the **service config** command to enable the loading of the specified configuration file at reboot time. Without this command, the communication server ignores the **boot host** command and uses the configuration information in nonvolatile memory. If the configuration information in nonvolatile memory 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. The second is the host configuration file containing commands that apply to one network server in particular.

Example

The following example sets the host filename to *wilma-confg* at address 192.31.7.19:

```
boot host /usr/local/tftpdir/wilma-confg 192.31.7.19
```

Related Commands

boot network
service config

boot network

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

> **boot network mop** *filename* [*mac-address*] [*interface*]
> **no boot network mop** *filename* [*mac-address*] [*interface*]

> **boot network** [**tftp** | **rcp**] *filename* [*ip-address*]
> **no boot network** [**tftp** | **rcp**] *filename* [*ip-address*]

Syntax Description

mop	Configures the communication server to download the configuration file from a network server using the Digital MOP protocol.
filename	Name of the file from which you want to load configuration commands. The default filename is *network-config*.
mac-address	(Optional) If **MOP** is specified, the MAC address of the network server on which the file resides. If the MAC address argument is not included, a broadcast message is sent to all MOP boot servers. The first server to indicate that it has the file will be the server from which the communication server gets the boot image.
interface	(Optional) If MOP is specified, the interface out which the communication server should send MOP requests to reach the server. The interface options are **async**, **dialer**, **ethernet**, **serial**, and **tunnel**. If the *interface* argument is not specified, a request will be sent on all interfaces that have MOP enabled, and the interface from which the first response is received will be used to load the software.
rcp	(Optional) Configures the communication server to download the configuration file from a network server using rcp. If omitted, defaults to **tftp**.
tftp	(Optional) Configures the communication server to download the configuration file from a network server using tftp. If omitted and **rcp** is not specified, defaults to **tftp**.
ip-address	(Optional) If **rcp** or **tftp** is specified, the IP address of the network server on which the compressed image file resides. If omitted, this value defaults to the IP broadcast address of 255.255.255.255.

Default

The default filename is *network-config*. The default transfer protocol type is **tftp**, if neither **tftp** nor **rcp** is specified.

Command Mode

Global configuration

Usage Guidelines

When booting from a network server, communication servers ignore routing information and static IP routes information. As a result, intermediate communication servers are responsible for handling rcp or tftp requests correctly. 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 communication server ignores the **boot network** command and uses the configuration information in nonvolatile memory. If the configuration information in nonvolatile memory is invalid or missing, the **service config** command is enabled automatically.

The network server attempts 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 rcp protocol requires that a client send the remote username on each rcp request to the network server. When the **boot network rcp** command is executed, the communication server software sends the host name as the both the remote and local usernames. The rcp protocol implementation searches for the configuration files to be used relative to the account directory of the remote username on the network server.

If you copy the system image to a PC used as a file server, the remote host computer must support the remote shell (rsh) protocol.

Caution For rcp, if you do not explicitly specify a remote username by issuing the **rcmd remote-username** command and the communication server host name is used, an account for the communication server host name must be defined on the destination server. If the network administrator of the destination server did not establish an account for the communication server host name, this command will not execute successfully.

If you copy the system image to a personal computer used as a file server, the remote host computer must support the remote shell protocol.

Examples

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

```
boot network bridge_9.1
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 131.108.1.111 as the IP address of the server on which the network configuration file resides.

```
boot network RCP bridge_9.1 131.108.1.111
service config
```

Related Commands

boot host
rcmd remote-username
service config

boot system

To change the filename of the system image that is loaded onto the communication server when it reboots, use the **boot system** global configuration command. Use the **no boot system** command to remove the name.

    boot system flash [*filename*]
    no boot system flash [*filename*]

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

    boot system rom
    no boot system rom

    boot system [**tftp** | **rcp**] *filename* [*ip-address*]
    no boot system [**tftp** | **rcp**] *filename* [*ip-address*]

    boot system flash [*device***:**][*partition-number***:**][*filename*]

    no boot system

Syntax Description

flash	Indicates that the communication server will be booted from Flash memory.
mop	Indicates that the communication server will be netbooted from a system image stored on a Digital MOP server.
rom	Indicates that the communication server will be booted from ROM.
rcp	(Optional) Indicates that the communication server will be netbooted from a system image acquired from a network server using rcp. If omitted, defaults to **tftp**.
tftp	(Optional) Indicates that the communication server will be booted from a system image stored on a network server using tftp. If omitted and **rcp** is not specified, defaults to **tftp**.
filename	(Optional with **flash**) Name of the configuration file from which you want to netboot. It is case sensitive.
ip-address	(Optional) IP address of the network server on which the image file resides. If omitted, this value defaults to the IP broadcast address of 255.255.255.255.
mac-address	(Optional) If MOP is used, the MAC address of the server on which the file resides. If the MAC address argument is not included, a broadcast message is sent to all MOP boot servers. The first server to indicate that it has the file will be the server from which the communication server gets the boot image.
interface	(Optional) Interface out which the communication server should send MOP requests to reach the MOP server. The interface options are **async**, **dialer**, **ethernet**, **serial**, and **tunnel**. If the *interface* argument is not specified, a request will be sent on all interfaces that have MOP enabled, and the interface from which the first response is received will be used to load the software.

device:	(Optional) Valid value is **flash**.
[*partition-number*:]	(Optional) Boots the communication server from Flash memory with the optional filename of the image you want loaded from the specified Flash partition. If you do not specify a filename, the first valid file in the specified partition of Flash memory is loaded.
filename	(Optional) Boots the communication server from Flash memory with the filename of the image you want loaded from the specified Flash partition. The filename is case sensitive. If you do not specify a filename, the first valid file in the specified partition of Flash memory will be loaded.

Default

If you do not specify a system image file with the **boot system** command, the communication server uses the configuration register settings to determine the default system image filename for netbooting. The communication server 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 (cisco*nn-cpu*). See the appropriate hardware installation guide for details on the configuration register and default filename. See also the command **config-register**. See also the "Syntax Description" section preceding this section.

The default transfer protocol type is **tftp** if neither **tftp** or **rcp** is specified.

Command Mode

Global configuration

Usage Guidelines

In order 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 communication server. Use the **boot system rom** command to specify use of the ROM system image as a backup to other **boot** commands in the configuration. You can enter the different types of **boot system** commands in any order. The communication server attempts to load from Flash memory first, then from a TFTP or MOP server, and finally from ROM. If you enter multiple boot commands of the same type—for example, if you enter two commands that instruct the communication server to boot from different network servers—the communication server tries them in the order they are entered.

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 The **no boot system** global configuration command disables all **boot system** configuration commands regardless of argument and keyword. Specifying the **flash** keyword or the *filename* argument with the **no boot system** command disables only the command specified by these arguments.

You can netboot from a compressed image. When booting from a network server, the image being booted and the running image must both fit into memory. Use compressed images to ensure that there is enough available memory to boot the communication server. You can produce a compressed 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.)

The rcp protocol requires that a client send the remote username on an rcp request to a server. When the **boot system rcp** command is executed, by default the communication server software sends the host name as the both the remote and local usernames. The rcp software searches for the system image to be booted from the remote server relative to the directory of the remote username.

Examples

The following example shows a list specifying two possible internetwork locations for a system image, with the ROM software being used as a backup. When the system image is booted from either of the internetwork locations, TFTP is used as the transport mechanism:

```
boot system cs3-rx.90-1 192.31.7.24
boot system cs3-rx.83-2 192.31.7.19
boot system rom
```

The following example boots the system boot relocatable image file *igs-bpx-l* from partition 2 of the Flash device.

```
boot system flash flash:2:igs-bpx-l
```

Related Commands

config-register
copy flash rcp
copy flash tftp
copy rcp flash
copy tftp flash
rcmd remote-username

config-register

To change the communication server configuration register settings, use the **config-register** global configuration command.

> **config-register** *value*

Syntax Description

value   Hexadecimal or decimal value that represents the 16-bit configuration register value you want to use the next time the communication server is restarted. The value range is from 0x0 to 0xFFFF (0 to 65535 in decimal).

Default

For the communication server models without Flash memory, the default is 0x101, which causes the communication server to boot from ROM and the Break key to be ignored. For communication server models with Flash memory, the default is 0x10F, which causes the communication server to boot from Flash memory and the Break key to be ignored.

Command Mode

Global configuration

Usage Guidelines

The lowest four bits of the configuration register (bits 3, 2, 1, and 0) form the boot field. The boot field determines if the communication server boots manually, from ROM, or from Flash or the network. Bit 8 controls the console Break key; when set to 1, it causes the Break key to be ignored. The remaining bits control other features of the communication server and are typically set to 0.

To change the boot field value and leave all other bits set to their default values, follow these guidelines:

- If you set the configuration register value to 0x100, you must boot the operating system manually with the **b** command.

- If you set the configuration register value to 0x101, the communication server boots using the default ROM software.

- If you set the configuration register to any value from 0x102 to 0x10F, the communication server uses the boot field value to form a default boot filename for netbooting.

For more information about the configuration register bit settings and default filenames, see the appropriate communication server hardware installation guide.

Example

In the following example, the configuration register is set to boot the system image from Flash memory:

```
config-register 0x010F
```

Related Commands

boot system
o
show version

configure

To enter global configuration mode, use the **configure** privileged EXEC command.

> **configure** {**terminal** | **memory** | **network**}

Syntax Description

terminal   Executes configuration commands from the terminal.

memory   Executes the configuration commands stored in nonvolatile memory.

network   Executes the configuration commands stored in a file on a server.

Default
None

Command Mode
Privileged EXEC

Usage Guidelines

If you do not specify **terminal**, **memory**, or **network**, the communication server prompts you for the source of configuration commands. After you enter the **configure** command, the system prompt changes from `cs-name#` to `cs-name(config)#`, indicating that you are in global configuration mode. To leave global configuration mode and return to the privileged EXEC prompt, press **Ctrl-Z**.

Note   The **configure net** command no longer clears line parameters, whether you select a host or network configuration file.

Examples

In the following example, the communication server is configured from the terminal:

```
cs# configure

Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
```

In the following example, the communication server is configured from the file *tokyo-confg* at IP address 131.108.2.155:

```
cs1# configure network

Host or network configuration file [host]?
IP address of remote host [255.255.255.255]? 131.108.2.155
Name of configuration file [tokyo-confg]?
Configure using tokyo-confg from 131.108.2.155? [confirm] y
Booting tokyo-confg from 131.108.2.155:!! [OK - 874/16000 bytes]
```

Related Commands

show configuration
configure overwrite
write memory
write terminal

configure overwrite

To load a configuration file directly into NVRAM, use the configure overwrite privileged EXEC command.

configure overwrite

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Privileged EXEC

Usage Guidelines

Use caution when entering the filename, because this command is not run through the parser. Also be careful not to load a file that is larger than NVRAM.

This command is useful if you are running an older version of software and are going to upgrade to a new IOS release. For example, if you have IOS Release 9.1 ROMs, you could save time by loading a 10.2 configuration file before you get the Release 10.2 software. That way, you will be ready to reboot when you receive the 10.2 software image.

This command also allows you to replace an entire old configuration, knowing that none of the old configuration will remain.

Example

```
doc-2500-cs# configure overwrite
Host or network configuration file [host]?
Address of remote host [255.255.255.255]? 131.108.1.111
Name of configuration file [doc-2500-cs-confg]? doc-2500-config
Configure using doc-2500-config from 131.108.1.111? [confirm]
Loading doc-2500-config...
```

Related Command

configure network

copy flash rcp

To copy a system image from Flash memory to a network server using rcp, use the **copy flash rcp** EXEC command.

> **copy flash rcp**

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

You can use the copy of the system image as a backup copy. You can also use it to verify that the copy in Flash memory is the same as on the original file on disk.

The rcp protocol requires that a client send the remote username on each rcp request to the server. When you issue the **copy flash rcp** command, by default the communication server software sends the remote username associated with the current TTY process, if that name is valid. If the TTY remote username is invalid, the communication server software uses the communication server host name as the both the remote and local usernames.

Note For UNIX systems, each physical device is represented in the file system. Terminals, or serial lines, are called TTY devices (which stands for teletype, the original UNIX terminal).

To specify a different remote username to be sent to the server, use the **rcmd remote-username** command. rcp copies the system image to the remote server relative to the directory of the remote username.

Caution The remote username must be associated with an account on the destination server. If you do not use the **rcmd remote-username** command to specify the name of a remote user associated with an account on the server, then the remote username associated with the current TTY process must be associated with an account on the server. If there is no username for the current TTY process, then the communication server host name must be associated with an account on the server. If the network administrator of the destination server did not establish accounts for the remote username used, this command will not execute successfully if a default remote username is used.

If you copy the system image to a PC used as a file server, the PC must support the remote shell (rsh) protocol.

Examples

The following example shows how to copy a system image from Flash memory to a network server using rcp:

```
cs# configure terminal
cs# rcmd remote-username netadmin1
Ctrl-Z
cs# copy flash rcp
System flash directory:
File name/status
 1 gsxx
[2076072 bytes used, 21080 bytes available]

Name of file to copy? gsxx
Address or name of remote host [UNKNOWN]? 131.108.1.111
File name to write to? gsxx
Verifying checksum for 'gsxx' (file # 1)...[OK]

Writing gsxx !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
cs#
```

The exclamation points (!) indicate that the copy process is taking place. Each exclamation point (!) indicates that ten packets have been transferred successfully.

The following example illustrates how to use this command when copying from a particular partition of Flash memory.

```
cs# copy flash rcp
System flash partition information:
Partition   Size      Used    Free      Bank-Size    State        Copy-Mode
    1       4096K     2048K   2048K     2048K        Read Only    RXBOOT-FLH
    2       4096K     2048K   2048K     2048K        Read/Write   Direct

[ Type ?<number> for partition directory; ? for full directory; q to abort]
Which partition? [default = 1]
```

The system will prompt if there are two or more partitions. If the partition entered is not valid, the process terminates. You have the option to enter a partition number, **?** for directory display of all partitions, or **?***number* for directory display of a particular partition. The default is the first partition.

```
System flash directory, partition 2:
File  Length   Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Address or name of remote host [ABC.CISCO.COM]?
Source file name?
```

The file will be copied from the partition given by the user earlier.

```
Destination file name [default = source name]?
Verifying checksum for 'master/igs-bfpx.100-4.3' (file # 1)...  OK
Copy 'master/igs-bfpx.100-4.3' from Flash to server
as 'master/igs-bfpx.100-4.3'? [yes/no] yes
```

Related Commands

boot system flash
copy rcp flash
rcmd remote-username

copy flash tftp

To copy a system image from Flash memory to a network server using TFTP, use the **copy flash tftp** EXEC command.

copy flash tftp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

You can use the copy of the system image to serve as a backup copy. You can also use it to verify that the copy in Flash memory is the same as on the original file on disk.

Example

The following example illustrates how to copy a system image from Flash memory to a network server using TFTP:

```
cs# copy flash tftp
IP address of remote host [255.255.255.255]? 101.2.13.110
Name of file to copy? gsxx
writing gsxx !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!copy complete
```

The following example illustrates how to use this command when copying from a particular partition of Flash memory.

```
cs# copy flash tftp
System flash partition information:
Partition   Size    Used    Free    Bank-Size   State       Copy-Mode
    1       4096K   2048K   2048K   2048K       Read Only   RXBOOT-FLH
    2       4096K   2048K   2048K   2048K       Read/Write  Direct

[ Type ?<number> for partition directory; ? for full directory; q to abort]
Which partition? [default = 1]
```

The system will prompt if there are two or more partitions. If the partition entered is not valid, the process terminates. You have the option to enter a partition number, **?** for directory display of all partitions, or **?***number* for directory display of a particular partition. The default is the first partition.

```
System flash directory, partition 2:
File  Length   Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Address or name of remote host [ABC.CISCO.COM]?
Source file name?
```

The file will be copied from the partition given by the user earlier.

```
Destination file name [default = source name]?
Verifying checksum for 'master/igs-bfpx.100-4.3' (file # 1)...  OK
Copy 'master/igs-bfpx.100-4.3' from Flash to server
as 'master/igs-bfpx.100-4.3'? [yes/no] yes
```

Related Commands

boot system flash
copy tftp flash

copy mop flash

To copy a system image using MOP into Flash memory, use the **copy mop flash** EXEC command.

copy mop flash

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The communication server prompts for the MOP filename. It provides an option to erase existing Flash memory before writing onto it. The entire copying process takes several minutes and will differ from network to network.

Before booting from Flash memory, verify that the checksum of the image in Flash memory matches the checksum listed in the README file that was distributed with the system software image. The checksum of the image in Flash memory is displayed at the bottom of the screen when you issue the **copy mop flash** command.

Caution   If the checksum value is not correct according to the value in the README file, do not reboot the communication server. Issue the **copy mop flash** command and compare the checksums again. If the checksum is repeatedly wrong, copy the original system software image back into Flash memory *before* you reboot the communication server from Flash memory. If you have a bad image in Flash memory and try to boot from Flash, the communication server will start the system image contained in ROM (assuming netbooting is not configured). If ROM does not contain a fully functional system image, the communication server might not function and will have to be reconfigured through a direct console port connection.

Example

The following example shows sample output of when copying a system image into a partition of Flash memory:

```
cs# copy mop flash
System flash partition information:
Partition   Size      Used     Free      Bank-Size    State        Copy-Mode
    1       4096K     2048K    2048K      2048K        Read Only    RXBOOT-FLH
    2       4096K     2048K    2048K      2048K        Read/Write   Direct

[ Type ?<no> for partition directory; ? for full directory; q to abort]
Which partition? [default = 2]
```

The system will prompt only if there are two or more read/write partitions or one read-only and one read/write partition and dual Flash bank support in boot ROMs. If the partition entered is not valid, the process terminates. You have the option to enter a partition number, **?** for directory display of all partitions, or **?***number* for directory display of a particular partition. The default is the first read/write partition.

If the partition is read-only and has dual Flash bank support in boot ROMs, the session continues as follows:

```
                        ****  NOTICE  ****
Flash load helper v1.0
This process will accept the copy options and then terminate
the current system image to use the ROM based image for the copy.
Routing functionality will not be available during that time.
If you are logged in via telnet, this connection will terminate.
Users with console access can see the results of the copy operation.
                        ---- ******** ----
Proceed? [confirm]
System flash directory, partition 1:
File  Length  Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Source file name? master/igs-bfpx-100.4.3
Destination file name [default = source name]?
```

The file will be copied into the partition given by the user earlier:

```
Loading master/igs-bfpx.100-4.3 from 131.108.1.111: !
Erase flash device before writing? [confirm]
Flash contains files. Are you sure? [confirm]
Copy 'master/igs-bfpx.100-4.3' from MOP server
as 'master/igs-bfpx.100-4.3' into Flash WITH erase? [yes/no] yes
```

If the partition is read-write, the session continues as follows:

```
System flash directory, partition 2:
File  Length  Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Source file name? master/igs-bfpx.100-4.3
Destination file name [default = source name]?
```

The file will be copied into the partition given by the user earlier:

```
Loading master/igs-bfpx.100-4.3 from 131.108.1.111: !
Erase flash device before writing? [confirm]
Flash contains files. Are you sure? [confirm]
Copy 'master/igs-bfpx.100-4.3' from MOP server
as 'master/igs-bfpx.100-4.3' into Flash WITH erase? [yes/no] yes
```

Related Commands

boot system flash
copy flash tftp
copy verify

copy rcp flash

To copy a system image from a network server into Flash memory using rcp, use the **copy rcp flash** EXEC command.

> **copy rcp flash**

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The communication server prompts for the address of the rcp server and rcp filename. It provides an option to erase existing Flash memory before writing onto it. The entire copying process takes several minutes and will differ from network to network.

Before booting from Flash memory, verify that the checksum of the image in Flash memory matches the checksum listed in the README file that was distributed with the system software image. The checksum of the image in Flash memory is displayed at the bottom of the screen when you issue the **copy rcp flash** command. The README file was copied to the server automatically when you installed the system software image.

> ⚠ **Caution** If the checksum value is not correct according to the value in the README file, do not reboot the communication server. Issue the **copy rcp flash** command and compare the checksums again. If the checksum is repeatedly wrong, copy the original system software image back into Flash memory *before* you reboot the communication server from Flash memory. If you have a bad image in Flash memory and try to boot from Flash, the communication server will start the system image contained in ROM (assuming netbooting is not configured). If ROM does not contain a fully functional system image, the communication server will not function and will have to be reconfigured through a direct console port connection.

The rcp protocol requires that a client send the remote username on each rcp request to the server. When you issue the **copy rcp flash** command, by default the communication server software sends the remote username associated with the current tty process, if that name is valid.

Note For UNIX systems, each physical device is represented in the file system. Terminals, or serial lines, are called TTY devices (which stands for teletype, the original UNIX terminal).

If the tty remote username is invalid, the communication server software uses the communication server host name as the both the remote and local usernames. To specify a different remote username to be sent to the network server, use the **rcmd remote-username** command. The rcp protocol copies the system image from the remote server relative to the directory of the remote username.

Caution The remote username must be associated with an account on the destination server. If you do not use the **rcmd remote-username** command to specify the name of a remote user associated with an account on the server, then the remote username associated with the current TTY process must be associated with an account on the server. If there is no username for the current TTY process, then the communication server host name must be associated with an account on the server. If the network administrator of the destination server did not establish accounts for the remote username used, this command will not execute successfully if a default remote username is used.

If you copy the system image from a PC used as a file server, the remote host computer must support the remote shell protocol.

Example

The following example copies a system image named *IJ09140z* from the *netadmin1* directory on the remote server named *SERVER1.CISCO.COM* with an IP address of 131.131.101.101 to the communication server's Flash memory. To ensure that enough Flash memory is available to accommodate the system image to be copied, the communication server software allows you to erase the contents of Flash memory first.

```
cs1# configure terminal
cs1# rcmd remote-username netadmin1
Ctrl-Z
cs1# copy rcp flash
System flash directory:
File name/status
 1 IJ09140Z
[2076072 bytes used, 21080 bytes available]

Address or name of remote host[UNKNOWN]? 131.131.101.101
Name of file to copy? IJ09140Z
Copy IJ09140z from SERVER1.CISCO.COM?[confirm]

Checking for file 'IJ09140Z' on SERVER1.CISCO.COM...[OK]

Erase flash device before writing?[confirm]
Are you sure?[confirm]
Erasing device...ezeeze...erased.

Connected to 131.131.101.101

Loading 2076007 byte file IJ09140Z:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!![OK]

Verifying checksum... (0x87FD)...[OK]
cs1#
```

The exclamation points (!) indicate that the copy process is taking place. Each exclamation point (!) indicates that ten packets have been transferred successfully.

The following example shows sample output of copying a system image named *IJ09140Z* into Flash memory:

```
cs# copy rcp flash
IP address or name of remote host [255.255.255.255]? server1
Name of rcp filename to copy into flash []? IJ09140Z
copy IJ09140Z from 131.131.101.101 into flash memory? [confirm] <Return>
xxxxxxxx bytes available for writing without erasure.
```

```
erase flash before writing? [confirm] <Return>
Clearing and initializing flash memory (please wait)####...
Loading from 101.2.13.110: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!... [OK - 324572/524212 bytes]
Verifying checksum...
VVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVV...
Flash verification successful. Length = 1204637, checksum = 0x95D9
```

The exclamation points (!) indicate the copy process. The series of Vs in the sample output indicates that a checksum verification of the image is occurring after the image is written to Flash memory.

The following example shows sample output when copying a system image into a partition of Flash memory.

```
cs# copy rcp flash
System flash partition information:
Partition   Size    Used    Free     Bank-Size   State       Copy-Mode
    1       4096K   2048K   2048K    2048K       Read Only   RXBOOT-FLH
    2       4096K   2048K   2048K    2048K       Read/Write  Direct

[ Type ?<no> for partition directory; ? for full directory; q to abort]
Which partition? [default = 2]
```

The system prompts only if there are two or more read/write partitions or one read-only and one read/write partition and dual Flash bank support in boot ROMs. If the partition entered is not valid, the process terminates. You have the option to enter a partition number, **?** for directory display of all partitions, or **?**_number_ for directory display of a particular partition. The default is the first read/write partition.

If the partition is read-only and has dual Flash bank support in boot ROM, the session continues as follows:

```
                     ****  NOTICE  ****
Flash load helper v1.0
This process will accept the copy options and then terminate
the current system image to use the ROM based image for the copy.
Routing functionality will not be available during that time.
If you are logged in via telnet, this connection will terminate.
Users with console access can see the results of the copy operation.
                     ---- ******** ----
Proceed? [confirm]
System flash directory, partition 1:
File  Length   Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Address or name of remote host [255.255.255.255]? 131.108.1.1
Source file name? master/igs-bfpx-100.4.3
Destination file name [default = source name]?
```

The file will be copied into the partition given by the user earlier:

```
Loading master/igs-bfpx.100-4.3 from 131.108.1.111: !
Erase flash device before writing? [confirm]
Flash contains files. Are you sure? [confirm]
Copy 'master/igs-bfpx.100-4.3' from TFTP server
as 'master/igs-bfpx.100-4.3' into Flash WITH erase? [yes/no] yes
```

If the partition is read-write, the session continues as follows:

```
System flash directory, partition 2:
File  Length   Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Address or name of remote host [255.255.255.255]? 131.108.1.1
```

```
Source file name? master/igs-bfpx.100-4.3
Destination file name [default = source name]?
```

The file will be copied into the partition given by the user earlier:

```
Accessing file 'master/igs-bfpx.100-4.3' on ABC.CISCO.COM...
Loading master/igs-bfpx.100-4.3 from 131.108.1.111: !
Erase flash device before writing? [confirm]
Flash contains files. Are you sure? [confirm]
Copy 'master/igs-bfpx.100-4.3' from TFTP server
as 'master/igs-bfpx.100-4.3' into Flash WITH erase? [yes/no] yes
```

Related Commands

boot system flash
copy flash rcp
copy verify
rcmd remote-username

copy rcp running-config

To copy a configuration file from a network server to the communication server using rcp and run that configuration, use the **copy rcp running-config** EXEC command.

> **copy rcp running-config**

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

(Cisco 2500 only.) Use this command to copy either a host configuration file or a network configuration file from a remote server to the communication server using rcp, load the configuration file into RAM, and run it on the communication server.

The communication server software allows you to specify the type of configuration file to be copied. Accept the default value of *host* to copy and load a host configuration file containing commands that apply to one network server in particular. Enter a value for *network* to copy and load a network configuration file containing commands that apply to all network servers on a network.

The rcp protocol requires that a client send the remote username on each rcp request to a server. When you issue the **copy rcp running-config** command, by default the communication server software sends the remote username associated with the current TTY process, if that name is valid. If the TTY remote username is invalid, the communication server software uses the communication server host name as the both the remote and local usernames. To specify a different remote username to be sent to the rcp server, use the **rcmd remote-username** command. rcp copies the configuration file from the remote server relative to the directory of the remote username.

Caution The remote username must be associated with an account on the destination server. If you do not use the **rcmd remote-username** command to specify the name of a remote user associated with an account on the server, then the remote username associated with the current TTY process must be associated with an account on the server. If there is no username for the current TTY process, then the communication server host name must be associated with an account on the server. If the network administrator of the destination server did not establish accounts for the remote username used, this command will not execute successfully if a default remote username is used.

Note For UNIX systems, each physical device is represented in the file system. Terminals, or serial lines, are called TTY devices (which stands for teletype, the original UNIX terminal).

If you copy the configuration file from a PC used as a file server, the PC must support the remote shell (rsh) protocol.

Example

The following example specifies a remote username of *netadmin1*. Then it copy and runs a host configuration file name *host1-confg* from the *netadmin1* directory on the remote server with an IP address of 131.131.101.101.

```
cs# configure terminal
cs# rcmd remote-username netadmin1
Ctrl-Z
cs# copy rcp running-config
Host or network configuration file [host]?
Address of remote host [255.255.255.255]? 131.131.101.101
Name of configuration file [cs-confg]? host1-confg
Configure using host1-confg from 131.131.101.101? [confirm]
Connected to 131.131.101.101
Loading 1112 byte file host1-confg:![OK]
cs#
%SYS-5-CONFIG: Configured from host1-config by rcp from 131.131.101.101
```

Related Commands

config running-config rcp
rcmd remote-username

copy rcp startup-config

To copy a configuration file from a network server to the nonvolatile memory of the communication server using rcp, use the **copy rcp startup-config** EXEC command.

copy rcp startup-config

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

(Cisco 2500 only.) Use this command to retrieve the commands stored in a configuration file on a server and write them to a file of the same name stored in nonvolatile memory on the communication server.

The communication server software allows you to specify the type of configuration file to be copied. Accept the default value of *host* to copy and store a host configuration file containing commands that apply to one network server in particular. Enter a value for *network* to copy and store a network configuration file containing commands that apply to all network servers on a network.

The rcp protocol requires that a client send the remote username on each rcp request to the server. When you issue the **copy rcp startup-config** command, by default the communication server software sends the remote username associated with the current tty process, if that name is valid. If the tty remote username is invalid, the communication server software uses the communication server host name as the both the remote and local usernames.

Note For UNIX systems, each physical device is represented in the file system. Terminals, or serial lines, are called TTY devices (which stands for teletype, the original UNIX terminal).

To specify a different remote username to be sent to the network server, use the **rcmd remote-username** command. The rcp protocol copies the configuration file from the remote server relative to the directory of the remote username.

Caution The remote username must be associated with an account on the destination server. If you do not use the **rcmd remote-username** command to specify the name of a remote user associated with an account on the server, then the remote username associated with the current TTY process must be associated with an account on the server. If there is no username for the current TTY process, then the communication server host name must be associated with an account on the server. If the network administrator of the destination server did not establish accounts for the remote username used, this command will not execute successfully if a default remote username is used.

If you copy the configuration file from a PC used as a file server, the PC must support the remote shell (rsh) protocol.

Example

The following example shows how to copy a configuration file to a Cisco 2500 system using rcp. This example specifies a remote username of *netadmin1*. Then it copies and stores a host configuration file *host2-confg* from the netadmin1 directory on the remote server with an IP address of 131.131.101.101:

```
cs2# configure terminal
cs2# rcmd remote-username netadmin1
Ctrl-Z
cs2# copy rcp startup-config
Host or network configuration file [host]?
Address of remote host [255.255.255.255]? 131.108.101.101
Name of configuration file[cs2-confg]? host2-confg
Configure using cs2-confg from 131.108.101.101?[confirm]
Connected to 131.108.101.101
Loading 1112 byte file cs2-confg:![OK]
[OK]
cs2#
%SYS-5-CONFIG_NV:Non-volatile store configured from cs2-config by rcp from
131.108.101.101
```

Related Commands

copy startup-config rcp
rcmd remote-username

copy running-config rcp

To copy the running configuration file from the communication server to a network server using rcp, use the **copy running-config rcp** EXEC command.

> **copy running-config rcp**

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

(Cisco 2500 only.) This command copies the current configuration file to a server on the network. The copy of the configuration file can serve as a backup copy. You are prompted for a destination host and filename.

The rcp protocol requires that a client send the remote username on each rcp request to the server. When you issue the **copy running-config rcp** command, by default the communication server software sends the remote username associated with the current tty process, if that name is valid. If the TTY remote username is invalid, the communication server software uses the communication server host name as the both the remote and local usernames.

Note For UNIX systems, each physical device is represented in the file system. Terminals, or serial lines, are called TTY devices (which stands for teletype, the original UNIX terminal).

To specify a different remote username to be sent to the network server, use the **rcmd remote-username** command. The rcp protocol copies the running configuration file to the remote server relative to the directory of the remote username that you specify.

Caution The remote username must be associated with an account on the destination server. If you do not use the **rcmd remote-username** command to specify the name of a remote user associated with an account on the server, then the remote username associated with the current TTY process must be associated with an account on the server. If there is no username for the current TTY process, then the communication server host name must be associated with an account on the server. If the network administrator of the destination server did not establish accounts for the remote username used, this command will not execute successfully if a default remote username is used.

If you copy the configuration file to a PC used as a file server, the PC must support the remote shell protocol.

To run this command, the communication server must contain Flash memory.

Example

The following example specifies a remote username of *netadmin1*. Then it copies the running configuration file, named *cs2-confg* to the *netadmin1* directory on the remote host with an IP address of 131.131.101.101:

```
cs2# configure terminal
cs2# rcmd remote-username netadmin1
Ctrl-Z
cs2# copy running-config rcp
Remote host[]? 131.131.101.101
Name of configuration file to write [cs2-confg]?
Write file cs2-confg on host 131.131.101.101?[confirm]
###![OK]
Connected to 131.131.101.101
cs2#
```

Related Commands

copy rcp running-config
rcmd remote-username

copy startup-config rcp

To copy a startup configuration file to a network server using rcp, use the **copy startup-config rcp** EXEC command.

copy startup-config rcp

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

Use this command to copy the contents of the configuration file in nonvolatile memory to a network server.

⚠ **Caution** The remote username must be associated with an account on the destination server. If you do not use the **rcmd remote-username** command to specify the name of a remote user associated with an account on the server, then the remote username associated with the current TTY process must be associated with an account on the server. If there is no username for the current TTY process, then the communication server host name must be associated with an account on the server. If the network administrator of the destination server did not establish accounts for the remote username used, this command will not execute successfully if a default remote username is used.

If you copy the configuration file to a PC used as a file server, the PC must support the remote shell protocol.

To run this command, the communication server must contain Flash memory.

Example

The following example shows how to copy a startup configuration file to a network server using rcp:

```
cs# configure terminal
cs# rcmd remote-username netadmin2
Ctrl-Z
cs# copy startup-config rcp
Remote host[]? 131.131.101.101
Name of configuration file to write [cs2-confg]?
Write file cs2-confg on host 131.131.101.101?[confirm]
![OK]
```

Related Commands

copy rcp startup-config
rcmd remote-username

copy tftp flash

To copy a system image into Flash memory using TFTP, use the **copy tftp flash** EXEC command.

copy tftp flash

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The communication server prompts for the address of the network server and TFTP filename. It provides an option to erase existing Flash memory before writing onto it. The entire copying process takes several minutes and will differ from network to network.

Before booting from Flash memory, verify that the checksum of the image in Flash memory matches the checksum listed in the README file that was distributed with the system software image. The checksum of the image in Flash memory is displayed at the bottom of the screen when you issue the **copy tftp flash** command. The README file was copied to the network server automatically when you installed the system software image.

⚠ **Caution** If the checksum value is not correct according to the value in the README file, do not reboot the communication server. Issue the **copy tftp flash** command and compare the checksums again. If the checksum is repeatedly wrong, copy the original system software image back into Flash memory *before* you reboot the communication server from Flash memory. If you have a bad image in Flash memory and try to boot from Flash, the communication server will start the system image contained in ROM (assuming netbooting is not configured). If ROM does not contain a fully functional system image, the communication server will not function and will have to be reconfigured through a direct console port connection.

Example

The following example shows sample output of copying a system image named *IJ09140Z* into Flash memory:

```
cs# copy tftp flash

IP address or name of remote host [255.255.255.255]? server1
Name of tftp filename to copy into flash []? IJ09140Z
copy IJ09140Z from 131.131.101.101 into flash memory? [confirm] <Return>
xxxxxxxx bytes available for writing without erasure.
erase flash before writing? [confirm] <Return>
Clearing and initializing flash memory (please wait)####...
Loading from 101.2.13.110: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!... [OK - 324572/524212 bytes]
Verifying checksum...
vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv...
Flash verification successful. Length = 1204637, checksum = 0x95D9
```

The exclamation points (!) indicate the copy process. The series of Vs in the sample output indicates that a checksum verification of the image is occurring after the image is written to Flash memory.

Different communication server platforms use different ways of indicating that Flash is being
cleared. The exclamation points (!) indicate the copy process. The series of Vs in the sample output
indicates that a checksum verification of the image is occurring after the image is written to Flash.
An O would indicate an out-of-order packet. A period (.) would indicate a timeout. The last line in
the sample configuration indicates that the copy was successful.

The following example shows sample output when copying a system image into a partition of Flash
memory.

```
cs# copy tftp flash
System flash partition information:
Partition   Size     Used    Free     Bank-Size   State       Copy-Mode
    1       4096K    2048K   2048K    2048K       Read Only   RXBOOT-FLH
    2       4096K    2048K   2048K    2048K       Read/Write  Direct

[ Type ?<no> for partition directory; ? for full directory; q to abort]
Which partition? [default = 2]
```

The system prompts only if there are two or more read/write partitions or one read-only and one
read/write partition and dual Flash bank support in boot ROMs. If the partition entered is not valid,
the process terminates. You have the option to enter a partition number, **?** for directory display of all
partitions, or **?***number* for directory display of a particular partition. The default is the first read/write
partition.

If the partition is read-only and has dual Flash bank support in boot ROM, the session continues as
follows:

```
                           ****  NOTICE  ****
Flash load helper v1.0
This process will accept the copy options and then terminate
the current system image to use the ROM based image for the copy.
Routing functionality will not be available during that time.
If you are logged in via telnet, this connection will terminate.
Users with console access can see the results of the copy operation.
                           ---- ******** ----
Proceed? [confirm]
System flash directory, partition 1:
File  Length   Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Address or name of remote host [255.255.255.255]? 131.108.1.1
Source file name? master/igs-bfpx-100.4.3
Destination file name [default = source name]?
```

The file is copied into the partition given by the user earlier:

```
Loading master/igs-bfpx.100-4.3 from 131.108.1.111: !
Erase flash device before writing? [confirm]
Flash contains files. Are you sure? [confirm]
Copy 'master/igs-bfpx.100-4.3' from TFTP server
as 'master/igs-bfpx.100-4.3' into Flash WITH erase? [yes/no] yes
```

If the partition is read-write, the session continues as follows:

```
System flash directory, partition 2:
File  Length   Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]
Address or name of remote host [255.255.255.255]? 131.108.1.1
Source file name? master/igs-bfpx.100-4.3
Destination file name [default = source name]?
```

The file is copied into the partition given by the user earlier:

```
Accessing file 'master/igs-bfpx.100-4.3' on ABC.CISCO.COM...
Loading master/igs-bfpx.100-4.3 from 131.108.1.111: !
Erase flash device before writing? [confirm]
Flash contains files. Are you sure? [confirm]
Copy 'master/igs-bfpx.100-4.3' from TFTP server
as 'master/igs-bfpx.100-4.3' into Flash WITH erase? [yes/no] yes
```

Related Commands

boot system flash
copy flash tftp
copy verify

copy verify

To verify the checksum of a system image in Flash memory, use the **copy verify** EXEC command.

copy verify

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **copy verify** command works on Cisco 2500 communication servers only.

Each system software image that is distributed on disk uses a single checksum for the entire image. This checksum is displayed only when the image is copied into Flash memory; it is *not* displayed when the image file is copied from one disk to another.

The README file (which is included with the image on the disk) lists the name, file size, and checksum of the image. Review the contents of the README file before loading or duplicating the new image so that you can verify the checksum when you copy it into the Flash memory or onto a network server.

To display the contents of Flash memory, use the **show flash** or **show flash all** command. The Flash content listing does not include the checksum of individual files. To recompute and verify the image checksum after the image is copied into Flash memory, use the **copy verify** command. When you enter the command, the screen prompts you for the filename to verify. By default, it prompts for the last file in Flash (most recent). Press **Return** to recompute the default file checksum or enter the filename of a different file at the prompt.

Example

The following example illustrates how to verify the checksum of a system image in Flash memory:

```
cs# copy verify

Name of file to verify [gsxx]?
Verifying via checksum...
vvvvvvvvvvvvvvvvvvvvvvvvvvvvv

Flash verification successful. Length = 1923712, checksum = 0xA0C1
cs#
```

Related Command

copy tftp flash

erase flash

To erase Flash memory, use the **erase flash** EXEC command.

erase flash

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

This command performs the same action as the **copy erase flash** command.

Example

The following example illustrates how to use this command:

```
cs# erase flash
System flash partition information:
Partition   Size    Used    Free    Bank-Size   State       Copy-Mode
    1       4096K   2048K   2048K   2048K       Read Only   RXBOOT-FLH
    2       4096K   2048K   2048K   2048K       Read/Write  Direct

[ Type ?<no> for partition directory; ? for full directory; q to abort]
Which partition? [default = 2]
```

The system will prompt only if there are two or more read/write partitions. If the partition entered is
not valid or is the read-only partition, the process terminates. You have the option to enter a partition
number, **?** for directory display of all partitions, or **?***number* for directory display of a particular
partition. The default is the first read/write partition.

```
System flash directory, partition 2:
File  Length    Name/status
  1   3459720   master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]

Erase flash device, partition 2? [confirm] <Return>
```

ip rarp-server

Use the **ip rarp-server** interface configuration command to allow the communication server to act as a Reverse Address Resolution Protocol (RARP) server. Use the **no** form of the command to restore the interface to the default of no RARP server support.

ip rarp-server *ip-address*
no ip rarp-server *ip-address*

Syntax Description

ip-address   IP address that is to be provided in the source protocol address field of the RARP response packet. Normally, this is set to whatever address you configure as the primary address for the interface.

Default
Disabled

Command Mode
Interface configuration

Usage Guidelines
This feature makes diskless booting of clients possible between network subnets where the client and server are on separate subnets.

RARP server support is configurable on a per interface basis, so that the communication server does not interfere with RARP traffic on subnets that do not need RARP assistance from the communication server.

The communication server answers incoming RARP requests only if both of the following conditions are met:

- The **ip rarp-server** command has been configured for the interface on which the request was received.

- There is a static entry found in the IP ARP table that maps the MAC address contained in the RARP request to an IP address.

Use the **show ip arp** EXEC command to display the contents of the IP ARP cache.

Sun Microsystems, Inc., makes use of RARP and UDP-based network services to facilitate network-based booting of SunOS on their workstations. By bridging RARP packets and using both the **ip helper-address** interface configuration command and the **ip forward-protocol** global configuration command, the communication server should be able to perform the necessary packet switching to enable booting of Sun workstations across subnets. Unfortunately, some Sun workstations assume that the sender of the RARP response, in this case the communication server, is the host the client can contact to TFTP load the bootstrap image. This causes the workstations to fail to boot.

By using the **ip rarp-server** feature, the communication server can be configured to answer these RARP requests, and the client machine should be able to reach its server by having its TFTP requests forwarded through the communication server that acts as the RARP server.

In the case of RARP responses to Sun workstations attempting to diskless boot, the IP address specified in the **ip rarp-server** interface configuration command should be the IP address of the TFTP server. In addition to configuring RARP service, the communication server must also be configured to forward UDP-based Sun portmapper requests to completely support diskless booting of Sun workstations. This can be accomplished using configuration commands of the form:

```
ip forward-protocol udp 111
interface interface name
ip helper-address target-address
```

RFC 903 documents the Reverse Address Resolution Protocol.

Examples

The following partial example configures the communication server to act as a RARP server. The communication server is configured to use the primary address of the specified interface in its RARP responses.

```
arp 128.105.2.5 0800.2002.ff5b arpa
interface ethernet 0
ip address 128.105.3.100 255.255.255.0
ip rarp-server 128.105.3.100
```

In the following example, the communication server is configured to act as a RARP server, with TFTP and portmapper requests forwarded to the Sun server:

```
! Allow the communication server to forward broadcast portmapper requests
ip forward-protocol udp 111
! Provide the communication server with the IP address of the diskless sun
arp 128.105.2.5 0800.2002.ff5b arpa
interface ethernet 0
! Configure the communication server to act as a RARP server, using the Sun Server's IP
! address in the RARP response packet.
ip rarp-server 128.105.3.100
! Portmapper broadcasts from this interface are sent to the Sun Server.
ip helper-address 128.105.3.100
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

ip forward-protocol [†]
ip helper-address [†]

mop device-code

To identify the type of device sending MOP sysid messages and request program messages, use the **mop device-code** global configuration command. Use the **no form of the** command to set the identity to the default value.

> **mop device-code** {**cisco** | **ds200**}
> **no mop device-code** {**cisco** | **ds200**}

Syntax Description

cisco   Denotes a Cisco device code.

ds200   Denotes a DECserver 200 device code.

Default
Cisco device code

Command Mode
Global configuration

Usage Guidelines
The sysid messages and request program messages use the identity information indicated by this command.

Example
The following example identifies a DECserver 200 device as sending MOP sysid and request program messages:

```
mop device-code ds200
```

Related Command
A dagger (†) indicates that the command is documented in another chapter.

mop sysid †

mop retransmit-timer

To configure the length of time the communication server waits before retransmitting boot requests to a MOP server, use the **mop retransmit-timer** global configuration command. Use the **no** form of the command to reinstate the default value.

mop retransmit-timer *seconds*
no mop retransmit-timer

Syntax Description

seconds	Sets the length of time, in seconds, that the communication server waits before retransmitting a message. The value is a number from 1 to 20.

Default

4 seconds

Command Mode

Global configuration

Usage Guidelines

By default, when the communication server transmits a request that requires a response from a MOP boot server and the server does not respond, the message will be retransmitted after 4 seconds. If the MOP boot server and communication server are separated by a slow serial link, it might take longer than 4 seconds for the communication server to receive a response to its message. Therefore, you might want to configure the communication server to wait longer than 4 seconds before retransmitting the message if you are using such a link.

Example

In the following example, if the MOP boot server does not respond within 10 seconds after the communication server sends a message, the server will retransmit the message:

```
mop retransmit-timer 10
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

mop device-code
mop retries
mop enabled [†]

mop retries

To configure the number of times a communication server will retransmit boot requests to a MOP server, use the **mop retries** global configuration command. Use the **no** form of the command to reinstate the default value.

> **mop retries** *count*
> **no mop retries**

Syntax Description

count	Indicates the number of times a communication server will retransmit a MOP boot request. The value is a number from 3 to 24.

Default

8 times

Command Mode

Global configuration

Example

In the following example, the communication server will attempt to retransmit a message to an unresponsive host 11 times before declaring a failure:

```
mop retries 11
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

mop device-code
mop retransmit-timer
mop enabled [†]

o

To list the value of the boot field (bits 0-3) in the configuration register, use the ROM monitor **o** command. To reset the value of the boot field so that the communication server boots from ROM, use the ROM monitor **o/r** command.

o
o/r

Syntax Description

This command has no arguments or keywords.

Default

Refer to the appropriate hardware installation guide for default values.

Command Mode

ROM monitor

Usage Guidelines

To get to the ROM monitor prompt at a communication server, use the **reload** EXEC command if the configuration register has a boot value of 0. (For systems with a software configuration register, a value can be included on the **o/r** command line.) Use the **i** command in conjunction with the **o/r** command to initialize the communication server. (The **i** command is documented in the hardware installation and maintenance publication for your product.) The **o/r** command resets the configuration register to 0x141, which disables the Break key, ignores the nonvolatile memory configuration, and boots the default system image from ROM.

Examples

The following is an example of the **o** command:

```
> o

Bit#    Configuration register option settings:
15      Diagnostic mode disabled
14      IP broadcasts do not have network numbers
13      Do not boot default ROM software if network boot fails
12-11   Console speed is 9600 baud
10      IP broadcasts with ones
09      Do not use secondary bootstrap
08      Break enabled
07      OEM disabled
06      Ignore configuration disabled
03-00   Boot to ROM monitor

>
```

The following is an example of the **o/r** and **i** commands used to reset and boot the default system image from ROM:

```
> o/r
> i
```

Related Command

config-register

partition flash

To partition Flash memory into two partitions, use the **partition flash** global configuration command. Use the **no** form of this command to undo partitioning, restoring Flash to one partition.

> **partition flash** *partitions* [*size1 size2*]
> **no partition flash**

Syntax Description

partitions	Number of partitions in Flash memory. Can be 1 or 2.
size1	(Optional) Size of the first partition in megabytes.
size2	(Optional) Size of the second partition in megabytes.

Default

Flash memory consists of one partition.

Command Mode

Global configuration

Usage Guidelines

To undo partitioning, use either the **partition flash 1** or **no partition flash** command.

Example

The following example creates two partitions of 4 MB each in Flash memory:

```
partition flash 2 4 4
```

rcmd domain-lookup

To enable Domain Name System (DNS) security for rcp and rsh, use the **rcmd domain-lookup** global configuration command. To bypass DNS security for rcp and rsh, use the **no** form of this command.

> **rcmd domain-lookup**
> **no rcmd domain-lookup**

Syntax Description

This command has no arguments or keywords.

Default

Enabled

Command Mode

Global configuration

Usage Guidelines

If you do not want to use DNS for rcmd queries and DNS has been enabled with the **ip domain-lookup** command, use the **no rcmd domain-lookup** command.

The **no rcmd domain-lookup** command turns off DNS lookups for rcp and rsh only. The **no ip domain-lookup** command takes precedence over the **rcmd domain-lookup** command. If you disable **ip domain-lookup** using the **no ip domain-lookup** command, DNS is bypassed for rcp and rsh even if you enable **rcmd domain-lookup**.

Example

The following example disables DNS security for rcp and rsh:

```
no rcmd domain-lookup
```

Related Command

ip domain-lookup

rcmd remote-host

To allow remote users to execute commands on the communication server, use the **rcmd remote-host** global configuration command to create an entry for the remote user in a local authentication database. Use the **no** form of the command to remove an entry for a remote user from the local authentication database.

> **rcmd remote-host** *local-username* {*ip-address* | *host*} *remote-username* [**enable**]
> **no rcmd remote-host** *local-username* {*ip-address* | *host*} *remote-username* [**enable**]

Syntax Description

local-username	Name of the user on the local communication server. You can specify the communication server host name as the username. This name needs to be communicated to the network administrator or the user on the remote system. To be allowed to remotely execute commands on the communication server, the remote user must specify this value correctly.
ip-address	IP address of the remote host from which the local communication server will accept remotely executed commands. Either the IP address or the host name is required.
host	Name of the remote host from which the local communication server will accept remotely executed commands. Either the host name or the IP address is required.
remote-username	Name of the user on the remote host from which the communication server will accept remotely executed commands.
enable	(Optional) Enables the remote user to execute privileged EXEC commands using rsh. This keyword does not apply to rcp.

Command Mode

Global configuration

Usage Guidelines

To allow a remote user to execute rcp or rsh commands on a local communication server, you must create an entry for the remote user in the local communication server's authentication database. You must also enable the communication server to act as an rsh or rcp server.

To enable the communication server to act as an rsh server, issue the **rsh-enable** command. To enable the communication server to act as an rcp server, issue the **rcp-enable** command. The communication server cannot act as a server for either of these protocols unless you explicitly enable the capacity.

A local authentication database, which is similar in concept and use to a UNIX *.rhosts* file, is used to enforce security on the communication server through access control. Each entry that you configure in the authentication database identifies the local user, the remote host, and the remote user. To permit a remote user of rsh to execute commands in privileged EXEC mode, specify the **enable** keyword.

An entry that you configure in the communication server authentication database differs from an entry in a UNIX *.rhosts* file in the following aspect. Because the *.rhosts* file on a UNIX system resides in the home directory of a local user account, an entry in a UNIX *.rhosts* file does not need to include the local username; the local username is determined from the user account. To provide equivalent support on a communication server configured as an rsh or rcp server, you specify the local username along with the remote host and remote username in each authentication database entry that you configure.

For a remote user to be able to execute commands on the communication server in its capacity as a server, the local username, host address or name, and remote username sent with the remote client request must match values configured in an entry in the local authentication file.

A remote client host should be registered with DNS. The communication server software uses DNS to authenticate the remote host's name and address. Because DNS can return several valid IP addresses for a host name, the communication server software checks the address of the requesting client against all of the IP addresses for the named host returned by DNS. If the address sent by the requester is considered invalid, that is, it does not match any address listed with DNS for the host name, then the communication server software will reject the remote-command execution request.

Note that if DNS support has been disabled for the local communication server, then the communication server cannot authenticate the host in this manner. In this case, the communication server software will send a broadcast request to attempt to gain access to DNS services on another server. If DNS services are unavailable, the communication server software will accept the request to remotely execute a command *only if* all three values sent with the request match exactly the values configured for an entry in the local authentication file.

To not check IP addresses against DNS, use the **no ip domain-lookup** command. This command disables DNS lookups.

Example

The following example allows the remote user *netadmin3* on a remote host with the IP address 131.131.101.101 to execute commands on *cs1* using the rsh protocol. For rsh, user *netadmin3* is allowed to execute commands in privileged EXEC mode:

```
rcmd remote-host cs1 131.131.101.101 netadmin3 enable
```

Related Commands

rcp-enable
rsh-enable
no ip domain-lookup

rcmd remote-username

To configure the remote username to be used when requesting a remote copy using rcp, use the **rcmd remote-username** global configuration command. To remove the remote username from the configuration, use the **no** form of the command.

> **rcmd remote-username** *username*
> **no rcmd remote-username** *username*

Syntax Description

username	Name of the remote user on the server. This name is used for rcp copy requests. All files and images to be copied are searched for or written relative to the directory of the remote user's account.

Command Mode

Global configuration

Usage Guidelines

Use this command also to configure the remote username for commands remotely executed from the communication server using rsh if you do not specify the **/user** keyword and argument on the **rsh** command.

The rcp protocol requires that a client send the remote username on an rcp request to the server. Use this command to specify the remote username to be sent to the server for an rcp copy request. When copying from the remote server, rcp searches for the system image or configuration file to be copied relative to the directory of the remote username. When copying to the remote server, rcp writes the system image or configuration file to be copied relative to the directory of the remote username.

Default

If you do not issue this command, the communication server software sends the remote username associated with the current TTY process, if that name is valid, for **rcp copy** commands. If the username for the current TTY process is not valid, the communication server software sends the host name as the remote username. For **rcp boot** commands, the communication server software sends the communication server host name by default.

Note For UNIX systems, each physical device is represented in the file system. Terminals, or serial lines, are called TTY devices (which stands for teletype, the original UNIX terminal).

The remote username must be associated with an account on the destination server.

Example

The following example shows how to configure the remote username to be used when requesting a remote copy using rcp:

```
rcmd remote-username netadmin1
```

Related Commands

boot network rcp
boot system rcp
copy flash rcp
copy rcp flash
copy rcp running-config
copy rcp startup-config
copy running-config rcp
copy startup-config rcp

rcp-enable

To configure the communication server to allow remote users to copy files to and from the communication server, use the **rcp-enable** global configuration command. Use the **no rcp-enable** command to disable a communication server that is enabled for rcp.

> **rcp-enable**
> **no rcp-enable**

Syntax Description

This command has no arguments or keywords.

Default

To ensure security, the communication server is *not* enabled for rcp by default. This is to protect against undesirable users copying the system image or configuration files without consent.

Command Mode

Global configuration

Usage Guidelines

To allow a remote user to execute rcp commands on the communication server, you must also create an entry for the remote user in the local communication server's authentication database.

The **no rcp-enable command** does not prohibit a local user from using rcp to copy system images and configuration files to and from the communication server.

Example

The following example shows how to use this command to enable the communication server for rcp:

```
cs# rcp-enable
```

Related Command

rcmd remote-host

reload

To reload the operating system, use the **reload** EXEC command.

reload

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **reload** command halts the system. If the system is set to restart on error, it reboots itself. The **reload** command is used after configuration information is entered into a file and saved into nonvolatile memory.

Example

The following example illustrates how to enter the **reload** command at the privileged EXEC prompt:

```
cs# reload
```

Related Command

write memory

rsh

To execute a command remotely on a remote rsh host, use the **rsh** EXEC command.

> **rsh** {*ip-address* | *host*} [**/user** *username*] *remote-command*

Syntax Description

ip-address	IP address of the remote host on which to execute the rsh command. Either the IP address or the host name is required.
host	Name of the remote host on which to execute the command. Either the host name or the IP address is required.
/user *username*	(Optional) Remote username. If you do not specify a remote username, the communication server software uses the configured remote username, if one exists. Otherwise, the communication server software uses the username associated with the current TTY, if it is a valid name. If this name is invalid, the communication server software uses the host name as the username.
remote-command	Command to be executed remotely. This is a required parameter. Unlike UNIX, the communication server software does not default to a remote login. Instead, the communication server provides telnet and connect services.

Command Mode

EXEC

Default

If you do not specify the **/user** keyword and argument, the communication server sends a default remote username unless you override the default by configuring a remote username. As the default value of the remote username, the communication server software sends the remote username associated with the current TTY process, if that name is valid. If the TTY remote username is invalid, the communication server software uses the communication server host name as the both the remote and local usernames.

Note For UNIX systems, each physical device is represented in the file system. Terminals, or serial lines, are called TTY devices (which stands for teletype, the original UNIX terminal).

Usage Guidelines

Use the rsh command to execute commands remotely. The host on which you remotely execute the command must support the remote shell (rsh) protocol, and the *.rhosts* files on the rsh host must include an entry that permits you to remotely execute commands on that host.

Example

The following example shows how to execute a command remotely on a remote rsh host:

```
cs# rsh mysys.cisco.com /u sharon ls -a
.
..
.alias
.cshrc
.emacs
.exrc
.history
.login
.mailrc
.newsrc
.oldnewsrc
.rhosts
.twmrc
.xsession
jazz
```

Related Command

rcmd remote-username

rsh-enable

To configure the communication server to allow remote users to execute commands on the communication server using rsh, use the **rsh-enable** global configuration command. Use the **no rsh-enable** command to disable a communication server that is enabled for rsh.

>**rsh-enable**
>**no rsh-enable**

Syntax Description

This command has no arguments or keywords.

Default

To ensure security, the communication server is *not* enabled for rsh by default.

Command Mode

Global configuration

Usage Guidelines

Use this command to enable the communication server to receive rsh requests from remote users. To allow a remote user to execute rsh commands on the communication server, you must also create an entry for the remote user in the local communication server's authentication database.

The **no rsh-enable command** does not prohibit a local user of the communication server from executing a command on other communication servers and UNIX hosts on the network using rsh.

Example

The following example shows how to use this command to enable the communication server as an rsh servers:

```
rsh-enable
```

Related Command

rcmd remote-host

service compress-config

To compress configuration files on communication servers that are equipped with nonvolatile memory, use the **service compress-config** global configuration command. To disable compression, use the **no** form of this command.

> **service compress-config**
> **no service compress-config**

Syntax Description

This command has no arguments or keywords.

Default

Disabled

Command Mode

Global configuration

Usage Guidelines

If the file compression completes successfully, the following message is displayed:

```
Compressing configuration from configuration-size to compressed-size
[OK]
```

If the boot ROMs do not recognize a compressed configuration, the following message is displayed:

```
Boot ROMs do not support NVRAM compression Config NOT written to NVRAM
```

If the file compression fails, the following message is displayed:

```
Error trying to compress nvram
```

One way to determine whether a configuration file will compress enough to fit into nonvolatile memory is to use a text editor to enter the configuration, then use the UNIX **compress** command to check the compressed size. To get a closer approximation of the compression ratio, use the UNIX command **compress -b12**.

Once the configuration file has been compressed, the communication server functions normally. A **show configuration** command would uncompress the configuration before displaying it. At boot time, the system would recognize that the configuration file was compressed, uncompress it, and proceed normally.

To disable compression of the configuration file, enter configuration mode and specify the **no service compress-config** command. Then enter the **write memory** command. The communication server displays an OK message if it is able to successfully write the uncompressed configuration to nonvolatile memory. Otherwise, the communication server displays an error message indicating that the configuration is too large to store. If the configuration file is larger than the physical nonvolatile memory, the following message is displayed:

```
###Configuration too large to fit uncompressed in NVRAM Truncate configuration? [confirm]
```

To truncate and save the configuration, type **Y**. To not truncate and not save the configuration, type **N**.

Example

In the following example, the configuration file is compressed:

```
service compress-config
```

Related Command

show configuration

service config

To enable automatic loading of configuration files from a network server, use the **service config** global configuration command. Use the **no** form of the command to restore the default.

> **service config**
> **no service config**

Syntax Description

This command has no arguments or keywords.

Default

Disabled, except on systems without nonvolatile memory or with invalid or incomplete information in nonvolatile memory. In these cases, autoloading of configuration files from a network server is enabled automatically.

Command Mode

Global configuration

Usage Guidelines

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 communication server to automatically configure the system from the file specified by the **boot host** or **boot network** command.

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 communication server uses the default configuration files. The default network configuration file is *network-confg*. The default host configuration file is *host-confg*, where *host* is the host name of the communication server. If the communication server cannot resolve its host name, the default host configuration file is *cs-confg*.

Example

In the following example, the communication server is configured to automatically load the default host configuration file:

```
boot host
service config
```

Related Commands

boot host
boot network

show async bootp

Use the **show async bootp** privileged EXEC command to display the parameters that have been configured for SLIP extended BOOTP requests.

> **show async bootp**

Syntax Description

This command has no arguments or keywords.

Command Mode

Privileged EXEC

Sample Display

The following is a sample output of the **show async bootp** command:

```
cs# show async bootp

The following extended data will be sent in BOOTP responses:

bootfile (for address 128.128.1.1) "pcboot"
bootfile (for address 131.108.1.111) "dirtboot"
subnet-mask 255.255.0.0
time-offset -3600
time-server 128.128.1.1
```

Table 3-2 describes significant fields shown in the display.

Table 3-2   show Async BOOTP Field Descriptions

Field	Description
bootfile... "pcboot"	Boot file for address 128.128.1.1 is named pcboot
subnet-mask 255.255.0.0	Subnet mask
time-offset -3600	Local time is one hour (3600 seconds) earlier than UTC time
time-server 128.128.1.1	Address of the time server for the network

Related Command

async-bootp

show configuration

Use the **show configuration** EXEC command to display the contents of the nonvolatile memory, if present and valid.

show configuration

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The nonvolatile memory stores the configuration information in the network server in text form as configuration commands. The **show configuration** command shows the version number of the software used when you last executed the **write memory** command.

Sample Display

The following is sample output from the **show configuration** command:

```
cs# show configuration

Using 5057 out of 32768 bytes
!
enable-password xxxx
service pad
!
boot system dross-system 131.108.13.111
boot system dross-system 131.108.1.111
!
exception dump 131.108.13.111
!
no ip ipname-lookup
!
decnet routing 13.1
decnet node-type area
decnet max-address 1023
!
interface Ethernet 0
ip address 131.108.1.1 255.255.255.0
ip helper-address 131.120.1.0
ip accounting
ip gdp
decnet cost 3
!
ip domain-name CISCO.COM
ip name-server 255.255.255.255
!
end
```

The following is partial sample output from the **show configuration** command when the configuration file has been compressed:

```
cs# show configuration
Using 21542 out of 65536 bytes, uncompressed size = 142085 bytes
!
version 9.22
service compress-config
!
hostname hosehead
!
boot system flash gs7-k.sthormod_clean
boot system rom
```

Related Commands

A dagger (†) indicates that the command is documented in another chapter.

configure
description †
service compress-config
write memory
write terminal

show flash

Use the **show flash** EXEC command to verify Flash memory. The **show flash** command displays the type of Flash memory present, any files that might currently exist in Flash memory, and the amounts of Flash memory used and remaining.

> **show flash** [**all** | **chips** | **detailed** | **err** | **partition** *number* [**all** | **chips** | **detailed** | **err**] |
> **summary**]

Syntax Description

all	(Optional) Shows complete information about Flash memory, including information about the individual ROM devices in Flash memory and the names and sizes of all system image files stored in Flash memory, including those that are invalidated.
chips	(Optional) Shows information per partition and per chip, including which bank the chip is in, its code, size, and name.
detailed	(Optional) Shows information per partition, including file length, address, name, Flash checksum, computer checksum, bytes used, bytes available, total bytes, and bytes of system Flash memory.
err	(Optional) Shows write or erase failures in the form of number of retries.
partition *number*	(Optional) Shows output for the specified partition number. If you specify the **partition** keyword, you must specify a partition number. You can use this keyword only when Flash memory has multiple partitions.
summary	(Optional) Shows summary information per partition, including the partition size, bank size, state, and method by which files can be copied into a particular partition. You can use this keyword only when Flash memory has multiple partitions.

Command Mode

EXEC

Sample Displays

The following is sample output from the **show flash** command on the Cisco 2500 series:

```
cs# show flash

System flash directory:

File     name/status
  0      ahp4/igs-bfpx.940705
  1      micro/eip1-0
  2      micro/sp1-3
  3      micro/trip1-1
  4      micro/hip1-0
  5      micro/fip1-1
  6      fsipucode
  7      spucode
  8      tripucode
  9      fipucode
 10      eipucode
 11      hipucode
 12      sipucode
 13      sp_q160-1
 14      ahp4/sp160-3 [deleted]
 15      ahp4/sp160-3
[4008468 bytes used, 185836 bytes available]
```

Table 3-3 describes the **show flash** display fields.

Table 3-3   Show Flash Field Descriptions

Field	Description
File	Number of file in Flash memory
name/status	Files that currently exist in Flash memory
bytes free	Amount of Flash memory remaining
[deleted]	Flag indicating that another file exists with the same name or that process has been aborted

As the display shows, the Flash memory can store and display multiple, independent software images for booting itself or for TFTP server software for other products. This feature is useful for storing default system software. These images can be stored in compressed format (but cannot be compressed by the communication server).

To eliminate any files from Flash memory (invalidated or otherwise) and free up all available memory space, the entire Flash memory must be erased; individual files cannot be erased from Flash memory.

The following is a sample output from the **show flash** command on a communication server that has Flash memory partitioned:

```
cs# show flash
System flash directory, partition 1:
  File  Length   Name/status
    1   3459720  master/igs-bfpx.100-4.3
  [3459784 bytes used, 734520 available, 4194304 total]
4096K bytes of processor board System flash (Read Only)

System flash directory, partition 2:
  File  Length   Name/status
    1   3459720  igs-kf
  [3459784 bytes used, 734520 available, 4194304 total]
```

```
    4096K bytes of processor board System flash (Read/Write)
```

In the following example, the security jumper is not installed and you cannot write to Flash memory until the security jumper is installed:

```
cs> show flash all
4096K bytes of flash memory on embedded flash (in RP1).
  security jumper(12V) is not installed,
flash memory is read-only.

file    offset length   name
00xDCD0    1903892    gs7-k [deleted]
10x1DEA24  1903912    gs7-k
 [329908/4194304 bytes free]
```

Table 3-4 describes the additional fields in the display.

Table 3-4   Show Flash All Fields for Partitioned Flash Memory

Field	Description
Partition	Partition number in Flash memory.
Size	Size of partition in bytes.
Used	Number of bytes used in partition.
Free	Number of bytes free in partition.
Bank-Size	Size of bank in bytes.
State	State of the partition. It can be one of the following values:
	• Read-Only—indicates the partition that is being executed from.
	• Read/Write—is a partition that can be copied to.
Copy-Mode	Method by which the partition can be copied to:
	• RXBOOT-FLH indicates copy via Flash Load Helper.
	• Direct indicates user can copy directly into Flash memory.
	• None indicates that it is not possible to copy into that partition.
Chip	Chip number.
Bank	Bank number.
Code	Code number.
Size	Size of chip.
Name	Name of chip.

The following is sample output for the **show flash chips** command on a communication server that has Flash memory partitioned.

```
cs# show flash chips
System flash partition 1:
4096K bytes of processor board System flash (Read ONLY)

  Chip    Bank    Code     Size       Name
   1       1      89A2     1024KB     INTEL 28F008SA
   2       1      89A2     1024KB     INTEL 28F008SA
   3       1      89A2     1024KB     INTEL 28F008SA
   4       1      89A2     1024KB     INTEL 28F008SA
Executing current image from System flash [partition 1]

System flash partition 2:
```

```
4096K bytes of processor board System flash (Read/Write)

Chip    Bank    Code     Size       Name
 1       2      89A2     1024KB     INTEL 28F008SA
 2       2      89A2     1024KB     INTEL 28F008SA
 3       2      89A2     1024KB     INTEL 28F008SA
 4       2      89A2     1024KB     INTEL 28F008SA
```

The following is sample output for the **show flash detailed** command on a communication server that has Flash memory partitioned.

```
cs# show flash detailed
System flash directory, partition 1:
File  Length    Name/status
        addr       fcksum  ccksum
 1   3224008  igs-kf.100
        0x40      0xEE91  0xEE91
[3224072 bytes used, 970232 available, 4194304 total]
4096K bytes of processor board System flash (Read/Write)

System flash directory, partition 2:
File  Length    Name/status
        addr       fcksum  ccksum
 1   3224008  igs-kf.100
        0x40      0xEE91  0xEE91
[3224072 bytes used, 970232 available, 4194304 total]
4096K bytes of processor board System flash (Read/Write)
```

The following is sample output for the **show flash summary** command on a communicaiton server that has Flash memory partitioned. The partition that indicates a state of "Read Only" is the partition that is being executed from.

```
cs# show flash summary
System flash partition information:
Partition   Size    Used     Free    Bank-Size    State        Copy-Mode
    1       4096K   2048K    2048K    2048K        Read Only    RXBOOT-FLH
    2       4096K   2048K    2048K    2048K        Read/Write   Direct
```

The following are possible values for Copy-Mode:

- RXBOOT-MANUAL—Copy manually by reloading to the boot ROM image.

- RXBOOT-FLH—Copy via Flash load helper

- Direct—Copy directly into flash memory

- None—Copy not allowed

show flh-log

To view the system console output generated during the Flash load helper operation, use the **show flh-log** EXEC command.

show flh-log

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

The **show flh-log** command enables a remote Telnet user performing the Flash upgrade without a console connection to retrieve console output when the Telnet connection is terminated by a switch to the ROM image. The output indicates what happened during the download, and is useful if the download fails.

Sample Output

The following is sample output of the **show-flh** command:

```
cs# show flh-log
%FLH: abc/igs-kf.914 from 131.108.1.111 to flash ...

System flash directory:
File   Length    Name/status
  1    2251320   abc/igs-kf.914

[2251384 bytes used, 1942920 available, 4194304 total]
Accessing file 'abc/igs-kf.914' on 131.108.1.111...
Loading from 131.108.13.111:

Erasing device...  ... erased
Loading from 131.108.13.111:
!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! [OK -
2251320/4194304 bytes]

Verifying checksum...  OK (0x97FA)
Flash copy took 79292 msecs
%FLH: Re-booting system after download
Loading abc/igs-kf.914 at 0x3000040, size = 2251320 bytes [OK]

F3: 2183364+67924+259584 at 0x3000060

                Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
 --More--

Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.
```

```
                    cisco Systems, Inc.
                    170 W. Tasman Drive
                    San Jose, California 95134

3000 Software (IGS-KF), Engineering Version 10.2(5.6)

Copyright (c) 1986-1995 by cisco Systems, Inc.
Compiled Thu 07-Jul-95 13:26 [enf 1048]


cisco 2500 (68030) processor (revision 0x00) with 4092K/2048K bytes of
memory.
Processor board serial number 00000000
DDN X.25 software, Version 2.0, NET2 and BFE compliant.
ISDN software, Version 1.0.
Bridging software.
Enterprise software set supported.  (0x0)
1 Ethernet/IEEE 802.3 interface.
2 Serial network interfaces.
 --More--

1 ISDN Basic Rate interface.
32K bytes of non-volatile configuration memory.

4096K bytes of processor board System flash (Read ONLY)
```

Related Command

copy tftp flash

show version

Use the **show version** EXEC command to display the configuration of the system hardware, the software version, the names and sources of configuration files, and the boot images.

   show version

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Sample Display

The following is sample output from the **show version** command from a Cisco 2500 series:

```
cs> show version

3000 Software (IGS-BFPX), Version 10.2
Copyright (c) 1986-1994 by cisco Systems, Inc.
Compiled Tue 05-Jul-94 16:14

System Bootstrap, Version (3.3), SOFTWARE

cs1 uptime is 6 days, 20 hours, 46 minutes
System restarted by reload
System image file is "achopra/igs-bfpx.940705", booted via flash

cisco 2500 (68030) processor (revision A) with 1024K/1024K bytes of memory.
Processor board serial number 01244583
X.25 software, Version 2.0, NET2, BFE and GOSIP compliant.
Bridging software.
SuperLAT software (copyright 1990 by Meridian Technology Corp).
Authorized for Enterprise software set.  (0x0)
1 Ethernet/IEEE 802.3 interface.
2 Serial network interfaces.
16 terminal lines.
32K bytes of non-volatile configuration memory.
4096K bytes of processor board System flash. (Read only mode)
Configuration register is 0x0
```

Table 3-5 describes significant fields shown in the display.

Table 3-5   Show Version Field Descriptions

Field	Description
software version	Information identifying the software by name and version number, including the date and time it was compiled. Always specify the complete version number when reporting a possible software problem. In the example output, the version number is 10.2.
System Bootstrap, Version…	Bootstrap version string.
Cs1 uptime is…	The amount of time the system has been up and running.
System restarted by…	Also displayed is a log of how the system was last booted, both as a result of normal system startup and of system error. For example, information can be displayed to indicate a bus error that is generally the result of an attempt to access a nonexistent address, as follows: `System restarted by bus error at PC 0xC4CA, address 0x210C0C0`
System image file is …	If the software was booted over the network, the IP address of the boot host is shown. If the software was loaded from onboard ROM, this line reads "running default software." In addition, the names and sources of the host and network configuration files are shown.
cisco 2500 (68030) processor…	The remaining output shows the hardware configuration and any nonstandard software options. The configuration register contents are displayed in hexadecimal notation.

The output of the **show version** EXEC command can also provide certain messages, such as bus error messages. If such error messages appear, report the complete text of this message to your technical support specialist.

tftp-server system

To specify that the communication server operate as a TFTP server, use the **tftp-server system** global configuration command. To remove a previously defined filename, use the **no tftp-server system** command with the appropriate filename and, optionally, the IP access list number.

tftp-server system *filename* [*access-list-number*]
no tftp-server system *filename* [*access-list-number*]
tftp-server system [**flash:**][*partition-number***:**]*filename* [*access-list-number*]

Syntax Description

filename	Name you give the communication server Flash file.
access-list-number	(Optional) IP access list number.
flash:	(Optional) Specifies TFTP server operation from the file in the first partition of Flash.
*partition-number***:**	(Optional) Specifies TFTP server operation from the file in the specified partition of Flash. If the partition number is not specified, the file in the first partition is used.
filename	Filename in the first or specified partition of Flash memory.

Default

Disabled

See also the "Syntax Description" and "Usage Guidelines" sections for more defaults.

Command Mode

Global configuration

Usage Guidelines

You can specify multiple filenames by repeating the **tftp-server system** command. The system sends a copy of the system image contained in ROM or one of the system images contained in Flash memory to any host that issues a TFTP read request with this filename.

The following algorithm is used when deciding whether to send the ROM or Flash image:

- If the specified *filename* exists in Flash memory, a copy of the Flash image is sent.

- If the specified *filename* is not found in Flash memory, the ROM image is sent.

Examples

Assuming there is a file in Flash named *version-9.0*, the following example causes the communication server to send, via TFTP, a copy of the Flash software when it receives a TFTP read request for the file *version-9.0*. The requesting host is checked against access list 22.

```
tftp-server system version-9.0 22
```

The following example causes the communication server to send, via TFTP, a copy of the file *flash:2:igs-bpx-1* when the requesting side specifies the name *flash:2:igs-bpx-l*:

```
tftp-server system flash:2:igs-bpx-l
```

Related Command

A dagger (†) indicates that the command is documented in another chapter.

access-list †

verify flash

To verify the checksums of files in Flash memory, use the **verify flash** EXEC command.

verify flash

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

This command performs the same action as the **copy verify flash** command.

Example

The following example illustrates how to use this command:

```
cs# verify flash
System flash partition information:
Partition   Size     Used    Free     Bank-Size  State       Copy-Mode
    1       4096K    2048K   2048K    2048K       Read Only   RXBOOT-FLH
    2       4096K    2048K   2048K    2048K       Read/Write  Direct

[ Type ?<no> for partition directory; ? for full directory; q to abort]
```

The system will prompt only if there are two or more read/write partitions. If the partition entered is not valid, the process terminates. You have the option to enter a partition number, **?** for directory display of all partitions, or **?***number* for directory display of a particular partition. The default is the first partition.

```
File  Length   Name/status
  1   3459720  master/igs-bfpx.100-4.3
[3459784 bytes used, 734520 available, 4194304 total]

Name of file to verify? master/igs-bfpx.100-4.3
Verifying checksum for 'master/igs-bfpx.100-4.3' (file # 1)...  OK
```

write erase

To erase the configuration information in nonvolatile memory, use the **write erase** EXEC command.

write erase

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Example

The following example illustrates how to erase the configuration in nonvolatile memory:

```
write erase
```

write memory

To copy the current configuration information to nonvolatile memory, use the **write memory** EXEC command.

write memory

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

Use the **write memory** command in conjunction with the **reload** command to restart the communication server with the configuration information stored in nonvolatile memory.

If you issue the **write memory** command from a bootstrap system image, a warning displays that the previous nonvolatile memory configuration will be overwritten and some of the configuration commands will be lost unless you answer no. This warning will not display if nonvolatile memory does not contain a valid configuration or if the previous configuration in nonvolatile memory was generated by a bootstrap system image.

Examples

The following example illustrates how to copy the current configuration information to nonvolatile memory:

```
cs# write memory
```

The following is an example of the warning the system provides if you are trying to save configuration information from bootstrap into the system:

```
cs(boot)# write memory

Warning: Attempting to overwrite an NVRAM configuration written by a full system image.
This bootstrap software does not support a full configuration command set. If you write
memory now, some configuration commands may be lost.
Overwrite the previous NVRAM configuration? [confirm]
```

Enter **no** to escape writing the configuration information to memory.

Related Commands

configure
reload
show configuration

write network

To copy the current configuration information to a network server, use the **write network** EXEC command.

> **write network**

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

This command copies the current configuration to a server host on the network. You are prompted for a destination host and filename.

Example

The following example illustrates how to begin the prompts for writing configuration information to a network host:

```
cs# write network

Remote host [0.0.0.0]? 131.108.1.111
Name of configuration file to write [cs-confg]?
Write file cs-confg on host 131.108.1.111? [confirm]
#
Writing cs-confg !! [OK]
cs#
```

write terminal

To display the current configuration information on the terminal, use the **write terminal** EXEC command.

write terminal

Syntax Description

This command has no arguments or keywords.

Command Mode

EXEC

Usage Guidelines

Use this command in conjunction with the **show configuration** command to compare the information in running memory to the information stored in nonvolatile memory.

Example

The following example illustrates how to display the current configuration information:

```
write terminal
```

Related Commands

configure
show configuration