R through setup

R through setup

R through setup

refuse-message

To define and enable a line-in-use message, use the **refuse-message** command in line configuration mode. To disable the message, use the **no**form of this command.

refuse-message *d message d*
no refuse-message

Syntax Description

d	Delimiting character of your choice--a pound sign (#), for example. You cannot use the delimiting character in the message.
message	Message text.

Command Default

Disabled (no line-in-use message is displayed).

Command Modes

Line configuration

Command History

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

Usage Guidelines

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character. You cannot use the delimiting character within the text of the message.

When you define a message using this command, the Cisco IOS software performs the following steps:

1. Accepts the connection.
2. Prints the custom message.
3. Clears the connection.

Examples

In the following example, line 5 is configured with a line-in-use message, and the user is instructed to try again later:

```
line 5
refuse-message  /The dial-out modem is currently in use.
Please try again later./
```

regexp optimize

To optimize the compilation of a regular expression access list, use the **regexp optimize** command in global configuration mode. To disable the configuration, use the **no** form of this command.

regexp optimize
no regexp optimize

Syntax Description

This command has no arguments or keywords.

Command Default

The command is enabled by default.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.
12.2(33)SRC	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SRC.
12.2(33)SXI	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SXI.
Cisco IOS XE Release 2.1	This command was implemented on the Cisco ASR 1000 Series Aggregation Services Routers.

Examples

The following example shows how to optimize the compilation of regular expression access list:

```
Router# configure terminal
Router(config)# regexp optimize
```

Related Commands

Command	Description
regexp (**profile map configuration**)	Creates an entry in a cache profile group that allows authentication and authorization matches based on a regular expression.

reload

To reload the operating system, use the **reload** command in privileged EXEC or diagnostic mode.

reload [{**/verify** | **/noverify**}] [{[**warm file**] [{**line** | **in** [{*hhh*:*mm* | *mmm* [*text*]}] | **at** *hh*:*mm* [*day month*] [*text*]}] | **reason** [*reason-string*] | **cancel**}]

Syntax Description

/verify	(Optional) Verifies the digital signature of the file that will be loaded onto the operating system.

/noverify	(Optional) Does not verify the digital signature of the file that will be loaded onto the operating system. **Note** This keyword is often issued if the **file verify auto** command is enabled, which automatically verifies the digital signature of all images that are copied.
warm	(Optional) Specifies warm rebooting.
file	(Optional) Specifies the image file for warm rebooting.
line	(Optional) Reason for reloading; the string can be from 1 to 255 characters long.
in *hhh* **:** *mm* | *mmm*	(Optional) Schedules a reload of the software to take effect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days.
text	(Optional) Reason for reloading; the string can be from 1 to 255 characters long.
at *hh* **:** *mm*	(Optional) Schedules a reload of the software to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight. The reload must take place within 24 days.
day	(Optional) Number of the day in the range from 1 to 31.
month	Month of the year.
reason *reason-string*	(Optional) Specifies a reason for reloading.
cancel	(Optional) Cancels a scheduled reload.

Command Modes

Privileged EXEC (#) Diagnostic (diag)

Command History

Release	Modification
10.0	This command was introduced.
12.2(14)SX	This command was modified. Support for this command was added for the Supervisor Engine 720.
12.3(2)T	This command was modified. The **warm** keyword was added.
12.2(18)S	This command was integrated into Cisco IOS Release 12.2(18)S. The /**verify**and **/noverify** keywords were added.
12.2(20)S	This command was modified. Support was added for the Cisco 7304 router. The Cisco 7500 series router in not supported in Cisco IOS Release 12.2(20)S.
12.0(26)S	This command was modified. The /**verify**and **/noverify** keywords were integrated into Cisco IOS Release 12.0(26)S.

Release	Modification
12.3(4)T	This command was modified. The /**verify**and **/noverify** keywords were integrated into Cisco IOS Release 12.3(4)T.
12.2(17d)SXB	This command was modified. Support for this command on the Supervisor Engine 2 was extended to Release 12.2(17d)SXB.
12.3(11)T	This command was modified. The **file** keyword and *url* argument were added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.0(1)M	This command was modified. The **reason** keyword and *reason-string*argument were added.
Cisco IOS XE Release 2.1	This command was introduced on the Cisco ASR 1000 Series Aggregation Services Router and was made available in diagnostic mode.

Usage Guidelines

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

You cannot reload from a virtual terminal if the system is not set up for automatic booting. This restriction prevents the system from using an image stored in the ROM monitor and taking the system out of the remote user's control.

If you modify your configuration file, the system prompts you to save the configuration. During a save operation, the system prompts whether you want to proceed with the save if the CONFIG_FILE variable points to a startup configuration file that no longer exists. If you respond "yes" in this situation, the system enters setup mode upon reload.

When you schedule a reload to occur at a later time (using the **in** keyword), it must take place within 24 days.

The **at** keyword can be used only if the system clock has been set on the router (either through Network Time Protocol [NTP], the hardware calendar, or manually). The time is relative to the configured time zone on the router. To schedule reloads across several routers to occur simultaneously, synchronize the time on each router with NTP.

When you specify the reload time using the **at** keyword, if you specify the month and day, the reload takes place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time), or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight. The reload must take place within 24 days.

To display information about a scheduled reload, use the **show reload** command.

The /verify and /noverify Keywords

If the /**verify** keyword is specified, the integrity of the image will be verified before it is reloaded onto a router. If verification fails, the image reload will not occur. Image verification is important because it assures the user that the image is protected from accidental corruption, which can occur at any time during transit, starting from the moment the files are generated by Cisco until they reach the user.

The **/noverify** keyword overrides any global automatic image verification that may be enabled via the **file verify auto** command.

The warm Keyword

If you issue the **reload** command after you have configured the **warm-reboot** global configuration command, a cold reboot will occur. Thus, if you want to reload your system, but do not want to override the warm reboot functionality, you should specify the **warm** keyword with the **reload** command. The warm reboot functionality allows a Cisco IOS image to reload without ROM monitor intervention. That is, read-write data is saved in RAM during a cold startup and restored during a warm reboot. Warm rebooting allows the router to reboot quicker than conventional rebooting (where control is transferred to ROM monitor and back to the image) because nothing is copied from flash to RAM.

Examples

The following example shows how to immediately reload the software on the router:

```
Router# reload
```

The following example shows how to reload the software on the router in 10 minutes:

```
Router# reload in 10
Router# Reload scheduled for 11:57:08 PDT Fri Apr 21 1996 (in 10 minutes)
Proceed with reload? [confirm]
```

The following example shows how to reload the software on the router at 1:00 p.m. on that day:

```
Router# reload at 13:00
Router# Reload scheduled for 13:00:00 PDT Fri Apr 21 1996 (in 1 hour and 2 minutes)
Proceed with reload? [confirm]
```

The following example shows how to reload the software on the router on April 21 at 2:00 a.m.:

```
Router# reload at 02:00 apr 21
Router# Reload scheduled for 02:00:00 PDT Sat Apr 21 1996 (in 38 hours and 9 minutes)
Proceed with reload? [confirm]
```

The following example shows how to cancel a pending reload:

```
Router# reload cancel
%Reload cancelled.
```

The following example shows how to perform a warm reboot at 4:00 a.m. on that day:

Router# **reload warm at 04:00**

The following example shows how to specify a reason for the reload:

Router# **reload reason** *reloaded with updated version*

The following example shows how to specify image verification via the **/verify** keyword before reloading an image onto the router:

```
Router# reload /verify
Verifying file integrity of bootflash:c7200-kboot-mz.121-8a.E
%ERROR:Signature not found in file bootflash:c7200-kboot-mz.121-8a.E.
Signature not present. Proceed with verify? [confirm]
Verifying file disk0:c7200-js-mz
.....................................................................
.....................................................Done!
Embedded Hash     MD5 :CFA258948C4ECE52085DCF428A426DCD
Computed Hash     MD5 :CFA258948C4ECE52085DCF428A426DCD
CCO Hash          MD5 :44A7B9BDDD9638128C35528466318183
```

```
Signature Verified
Proceed with reload? [confirm]n
```

Related Commands

Command	Description
copy system:running-config nvram:startup-config	Copies any file from a source to a destination.
file verify auto	Enables automatic image verification.
show reload	Displays the reload status on the router.
warm-reboot	Enables router reloading with reading images from storage.

remote command

To execute a Cisco 7600 series router command directly on the switch console or a specified module without having to log into the Cisco 7600 series router first, use the **remote command** command in privileged EXEC mode.

remote command {**module** *num* | **standby-rp** | **switch**} *command*

Syntax Description

module *num*	Specifies the module to access; see the "Usage Guidelines" section for valid values.
standby-rp	Specifies the standby route processor.
switch	Specifies the active switch processor.
command	Command to be executed.

Command Default

This command has no default settings.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(14)SX	Support for this command was introduced on the Supervisor Engine 720.
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was extended to Release 12.2(17d)SXB.
12.2(18)SXD	The **standby-rp** keyword was added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

The **module** *num* keyword and argument designate the module number. Valid values depend on the chassis that is used. For example, if you have a 13-slot chassis, valid values are from 1 to 13. The **module** *num* keyword and argument are supported on DFC-equipped modules and the standby supervisor engine only.

When you execute the **remote command switch**command, the prompt changes to Switch-sp#.

This command is supported on DFC-equipped modules and the supervisor engine only.

This command does not support command completion, but you can use shortened forms of the command (for example, entering **sh** for **show**).

Examples

This example shows how to execute the **show calendar** command from the standby route processor:

```
Router#
remote command standby-rp show calendar
Switch-sp#
09:52:50 UTC Mon Nov 12 2001
Router#
```

Related Commands

Command	Description
remote login	Accesses the Cisco 7600 series router console or a specific module.

remote login

To access the Cisco 7600 series router console or a specific module, use the **remote login** command in privileged EXEC mode.

remote login {**module** *num* | **standby-rp** | **switch**}

Syntax Description

module *num*	Specifies the module to access; see the "Usage Guidelines" section for valid values.
standby-rp	Specifies the standby route processor.
switch	Specifies the active switch processor.

Command Default

This command has no default settings.

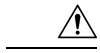
Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(14)SX	Support for this command was introduced on the Supervisor Engine 720.
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was extended to Release 12.2(17d)SXB.
12.2(18)SXD	This command was changed to include the **standby-rp** keyword.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines



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

The **module** *num* keyword and argument designate the module number. Valid values depend on the chassis that is used. For example, if you have a 13-slot chassis, valid values are from 1 to 13. The **module** *num* keyword and argument are supported on DFC-equipped modules and the standby supervisor engine only.

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

When you execute the **remote login standby-rp** command, the prompt changes to Router-sdby#.

When you execute the **remote login switch** command, the prompt changes to Switch-sp#.

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

There are two ways to end the session:

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

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

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

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

Examples

This example shows how to perform a remote login to a specific module:

```
Router# remote login module 1
Trying Switch ...
Entering CONSOLE for Switch
Type "^C^C^C" to end this session
Switch-sp#
```

This example shows how to perform a remote login to the Cisco 7600 series router processor:

```
Router# remote login switch
Trying Switch ...
Entering CONSOLE for Switch
Type "^C^C^C" to end this session
Switch-sp#
```

This example shows how to perform a remote login to the standby route processor:

```
Router# remote login standby-rp
Trying Switch ...
Entering CONSOLE for Switch
Type "^C^C^C" to end this session
Router-sdby#
```

Related Commands

Command	Description
attach	Connects to a specific module from a remote location.

remote-span

To configure a virtual local area network (VLAN) as a remote switched port analyzer (RSPAN) VLAN, use the **remote-span** command in config-VLAN mode. To remove the RSPAN designation, use the **no** form of this command.

remote-span
no remote-span

Syntax Description

This command has no arguments or keywords.

Command Default

This command has no default settings.

Command Modes

Config-VLAN mode

Command History

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

Usage Guidelines

This command is not supported in the VLAN database mode.

You can enter the **show vlan remote-span** command to display the RSPAN VLANs in the Cisco 7600 series router.

Examples

This example shows how to configure a VLAN as an RSPAN VLAN:

```
Router(config-vlan)# remote-span
Router(config-vlan)
```

This example shows how to remove the RSPAN designation:

```
Router(config-vlan)# no remote-span
Router(config-vlan)
```

Related Commands

Connect	Description
show vlan remote-span	Displays a list of RSPAN VLANs.

rename

To rename a file in a Class C Flash file system, use the **rename** command in EXEC, privileged EXEC, or diagnostic mode.

rename *url1 url2*

Syntax Description

url1	The original path and filename.
url2	The new path and filename.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Diagnostic (diag)

Command History

Release	Modification
11.3 AA	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE Release 2.1	This command was introduced on the Cisco ASR 1000 Series Router and was made available in diagnostic mode.

Usage Guidelines

This command is valid only on Class C Flash file systems.

Examples

In the following example, the file named Karen.1 is renamed test:

```
Router# dir
Directory of disk0:/Karen.dir/

  0  -rw-           0   Jan 21 1998 09:51:29  Karen.1
  0  -rw-           0   Jan 21 1998 09:51:29  Karen.2
  0  -rw-           0   Jan 21 1998 09:51:29  Karen.3
  0  -rw-           0   Jan 21 1998 09:51:31  Karen.4
243  -rw-         165   Jan 21 1998 09:53:17  Karen.cur

340492288 bytes total (328400896 bytes free)
Router# rename disk0:Karen.dir/Karen.1 disk0:Karen.dir/test

Router# dir
Directory of disk0:/Karen.dir/

  0  -rw-           0   Jan 21 1998 09:51:29  Karen.2
  0  -rw-           0   Jan 21 1998 09:51:29  Karen.3
  0  -rw-           0   Jan 21 1998 09:51:31  Karen.4
243  -rw-         165   Jan 21 1998 09:53:17  Karen.cur
  0  -rw-           0   Apr 24 1998 09:49:19  test

340492288 bytes total (328384512 bytes free)
```

request consent-token accept-response shell-access

To submit the Consent Token response to a previously generated challenge, use the **request consent-token accept-response shell-access** command.

request consent-token accept-response shell-access *response-string*

Syntax Description

Syntax	Description
response-string	Specifies the character string representing the response.

Command Modes Privileged EXEC mode (#)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.11.1	This command was introduced.

Usage Guidelines You must enter the response string within 30 minutes of challenge generation. If it is not entered, the challenge expires and a new challenge must be requested.

Example

The following is sample output from the **request consent-token accept-response shell-access** *response-string* command:

```
Device# request consent-token accept-response shell-access
[redacted]
% Consent token authorization success
*Jan 18 02:51:37.807: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (authentication success:
 Shell access 0).
```

request consent-token generate-challenge shell-access

To generate a Consent Token challenge for system shell access, use the **request consent-token generate-challenge shell-access** command.

request consent-token generate-challenge shell-access auth-timeout *time-validity-slot*

Syntax Description

Syntax	Description
auth-timeout *time-validity-slot*	Specifies the time slot in minutes for which shell-access is requested.

Command Modes Privileged EXEC mode (#)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.11.1	This command was introduced.

Usage Guidelines When the requested time-slot for system shell expires, the session gets terminated automatically.

The maximum authorization timeout for system shell access is seven days.

Example

The following is sample output from the **request consent-token generate-challenge shell-access auth-timeout** *time-validity-slot* command:

```
Device# request consent-token generate-challenge shell-access auth-timeout 900
zSSdrAAAAQEBAAQAAAABAgAEAAAAAAMACH86csUhmD1OBAAQ0FvdJ7CxqRYUeoD7B4AwW7QUABAAAAG8GAAhDVEFfREVNTwcAGENUQV9ERU1PX0NUQV9TSUdOSU5HX0tFWQgAC0M5ODAwLUNMUs5CQAL0VpQUEVESE5KRkI=
Device#
*Jan 18 02:47:06.733: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (challenge generation
attempt: Shell access 0).
```

request consent-token terminate-auth

To terminate the Consent Token based authorization to system shell, use the **request consent-token terminate-auth** command.

request consent-token terminate-auth

Command Modes

Privileged EXEC mode (#)

Command History

Release	Modification
Cisco IOS XE Gibraltar 16.11.1	This command was introduced.

Usage Guidelines

In system shell access scenario, exiting the shell does not terminate authorization until the authorization timeout occurs.

We recommend that you force terminate system shell authorization by explicitly issuing the **request consent-token terminate-auth** command once the purpose of system shell access is complete.

If the current authentication is terminated using the **request consent-token terminate-auth** command, the user will have to repeat the authentication process to gain access to system shell.

Example

The following is sample output from the **request consent-token terminate-auth** command:

```
Device# request consent-token terminate-auth shell-access
% Consent token authorization termination success

Device#
*Mar 13 01:45:39.197: %CTOKEN-6-AUTH_UPDATE: Consent Token Update (terminate authentication:
 Shell access 0).
Device#
```

request platform software package describe file

To gather descriptive information about an individual module or a Cisco IOS-XE image file, use the **request platform software package describe file**command in privileged EXEC or diagnostic mode.

request platform software package describe file *URL* [**detail**] [**verbose**]

Syntax Description

URL	Specifies the URL to the file. The *URL* contains the file system, directories, and the filename.

detail	Specifies detailed output.
verbose	Displays verbose information, meaning all information that can be displayed on the console about the file will be displayed.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC (#)

Diagnostic (diag)

Command History

Release	Modification
IOS XE Release 2.1	This command was introduced.

Usage Guidelines

This command can only be used to gather information on individual module and Cisco IOS-XE image files. Using this command to collect information on any other file will generate output, but the generated output is useless.

The output of this command can be used for the following functions:

- To confirm the individual module files that are part of a Cisco IOS-XE image.
- To confirm whether or not a file is bootable.
- To confirm the contexts in which a file must be reloaded or booted.
- To confirm whether or not a file is corrupted.
- To confirm file and header sizes, build dates, and various other general information.

Examples

In the following example, this command is entered to gather information about an individual SIP Base module file on the bootflash: file system.

```
Router# request platform software package describe file
bootflash:asr1000rp1-sipbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
Package: asr1000rp1-sipbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Size: 36954316
  Timestamp: 2007-12-05 15:36:27 UTC
  Canonical path:
/bootflash/asr1000rp1-sipbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg

  Raw disk-file SHA1sum:
    3ee37cdbe276316968866b16df7d8a5733a1502e

  Computed SHA1sum:
    f2db80416a1245a5b1abf2988088860b38ce7898
  Contained SHA1sum:
    f2db80416a1245a5b1abf2988088860b38ce7898
  Hashes match. Package is valid.

  Header size:     204 bytes
  Package type:    10000
  Package flags:   0
  Header version:  0
```

```
  Internal package information:
    Name: cc
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: sipbase
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is bootable on SIP when specified
  by packages provisioning file.
```

In the following example, this command is used to gather information about a Cisco IOS-XE image on the bootflash: file system.

```
Router# request platform software package describe file
bootflash:ASR1000rp1-advipservicesk9.01.00.00.12-33.XN.bin
Package: ASR1000rp1-advipservicesk9.01.00.00.12-33.XN.bin
  Size: 218783948
  Timestamp: 2007-12-04 17:14:09 UTC
  Canonical path: /bootflash/ASR1000rp1-advipservicesk9.01.00.00.12-33.XN.bin

  Raw disk-file SHA1sum:
    d2999fc7e27e01344903a42ffacd62c156eba4cc

  Computed SHA1sum:
    5f8cda8518d01d8282d80ecd34f7715783f4a813
  Contained SHA1sum:
    5f8cda8518d01d8282d80ecd34f7715783f4a813
  Hashes match. Package is valid.

  Header size:     204 bytes
  Package type:    30000
  Package flags:   0
  Header version:  0

  Internal package information:
    Name: rp_super
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: advipservicesk9
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is bootable from media and tftp.
  Package contents:

  Package: asr1000rp1-espbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
    Size: 52072652
    Timestamp: 2007-12-04 13:33:13 UTC

    Raw disk-file SHA1sum:
      f1aad6d687256aa327a4efa84deab949fbed12b8

    Computed SHA1sum:
      15502fd1b8f9ffd4af4014ad4d8026c837929fe6
    Contained SHA1sum:
      15502fd1b8f9ffd4af4014ad4d8026c837929fe6
```

```
  Hashes match. Package is valid.

  Header size:     204 bytes
  Package type:    20000
  Package flags:   0
  Header version:  0

  Internal package information:
    Name: fp
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: espbase
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is bootable on ESP when specified
  by packages provisioning file.

Package: asr1000rp1-rpaccess-k9.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Size: 21844172
  Timestamp: 2007-12-04 13:33:01 UTC

  Raw disk-file SHA1sum:
    025e6159dd91cef9d254ca9fff2602d8ce065939

  Computed SHA1sum:
    ea1b358324ba5815b9ea623b453a98800eae1c78
  Contained SHA1sum:
    ea1b358324ba5815b9ea623b453a98800eae1c78
  Hashes match. Package is valid.

  Header size:     204 bytes
  Package type:    30004
  Package flags:   0
  Header version:  0

  Internal package information:
    Name: rp_security
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: rpaccess-k9
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is not bootable.

Package: asr1000rp1-rpbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Size: 21520588
  Timestamp: 2007-12-04 13:33:06 UTC

  Raw disk-file SHA1sum:
    432dfa61736d8a51baefbb2d70199d712618dcd2

  Computed SHA1sum:
    83c0335a3adcea574bff237a6c8640a110a045d4
  Contained SHA1sum:
    83c0335a3adcea574bff237a6c8640a110a045d4
  Hashes match. Package is valid.
```

```
  Header size:     204 bytes
  Package type:    30001
  Package flags:   0
  Header version:  0

  Internal package information:
    Name: rp_base
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: rpbase
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is bootable on RP when specified
  by packages provisioning file.

Package: asr1000rp1-rpcontrol.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Size: 24965324
  Timestamp: 2007-12-04 13:33:08 UTC

  Raw disk-file SHA1sum:
    eb964b33d4959c21b605d0989e7151cd73488a8f

  Computed SHA1sum:
    19b58886f97c79f885ab76c1695d1a6f4348674e
  Contained SHA1sum:
    19b58886f97c79f885ab76c1695d1a6f4348674e
  Hashes match. Package is valid.

  Header size:     204 bytes
  Package type:    30002
  Package flags:   0
  Header version:  0

  Internal package information:
    Name: rp_daemons
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: rpcontrol
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is not bootable.

Package: asr1000rp1-rpios-advipservicesk9.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg

  Size: 48515276
  Timestamp: 2007-12-04 13:33:13 UTC

  Raw disk-file SHA1sum:
    bc13462d6a4af7a817a7346a44a0ef7270e3a81b

  Computed SHA1sum:
    f1235d703cc422e53bce850c032ff3363b587d70
  Contained SHA1sum:
    f1235d703cc422e53bce850c032ff3363b587d70
  Hashes match. Package is valid.
```

```
  Header size:     204 bytes
  Package type:    30003
  Package flags:   0
  Header version:  0

  Internal package information:
    Name: rp_iosd
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: rpios-advipservicesk9
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is not bootable.

Package: asr1000rp1-sipbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Size: 36954316
  Timestamp: 2007-12-04 13:33:11 UTC

  Raw disk-file SHA1sum:
    3ee37cdbe276316968866b16df7d8a5733a1502e

  Computed SHA1sum:
    f2db80416a1245a5b1abf2988088860b38ce7898
  Contained SHA1sum:
    f2db80416a1245a5b1abf2988088860b38ce7898
  Hashes match. Package is valid.

  Header size:     204 bytes
  Package type:    10000
  Package flags:   0
  Header version:  0

  Internal package information:
    Name: cc
    BuildTime: 2007-12-04_05.24
    ReleaseDate: Tue 04-Dec-07 01:00
    RouteProcessor: rp1
    Platform: ASR1000
    User: mcpre
    PackageName: sipbase
    Build: v122_33_xn_asr_rls0_throttle_20071204_051318


  Package is bootable on SIP when specified
  by packages provisioning file.

Package: asr1000rp1-sipspa.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Size: 19933388
  Timestamp: 2007-12-04 13:33:06 UTC

  Raw disk-file SHA1sum:
    44b6d15cba31fb0e9b27464665ee8a24b92adfd2

  Computed SHA1sum:
    b1d5faf093b183e196c7c8e1023fe1f7aafdd36d
  Contained SHA1sum:
    b1d5faf093b183e196c7c8e1023fe1f7aafdd36d
  Hashes match. Package is valid.
```

```
Header size:     204 bytes
Package type:    10001
Package flags:   0
Header version:  0

Internal package information:
  Name: cc_spa
  BuildTime: 2007-12-04_05.24
  ReleaseDate: Tue 04-Dec-07 01:00
  RouteProcessor: rp1
  Platform: ASR1000
  User: mcpre
  PackageName: sipspa
  Build: v122_33_xn_asr_rls0_throttle_20071204_051318


Package is not bootable.
```

Related Commands

Command	Description
request platform software package install file	Upgrades an individual package or a superpackage file.

request platform software package expand file

To extract the individual modules from a Cisco IOS-XE image, use the **request platform software package expand file**command in privileged EXEC or diagnostic mode.

request platform software package expand file *source-URL* [**to** *destination-URL*] [**force**] [**verbose**] [**wipe**]

Syntax Description

source-URL	Specifies the URL to the Cisco IOS-XE file that stores the contents that will be extracted.
to destination-URL	Specifies the destination URL where the files that were extracted from the Cisco IOS-XE file are left after the operation is complete. If this option is not entered, the Cisco IOS-XE image file contents are extracted onto the same directory where the Cisco IOS-XE image file is currently stored.
force	(Optional) Specifies that the operation will be forced, meaning that the upgrade will proceed despite any warning messages.
verbose	(Optional) Displays verbose information, meaning all output that can be displayed on the console during the process will be displayed.
wipe	(Optional) Erases all content on the destination snapshot directory before extracting the files and placing them on the snapshot directory.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#)

Diagnostic Mode (diag)

Command History

Release	Modification
IOS XE Release 2.1	This command was introduced.

Usage Guidelines

This command only extracts individual module files and a provisioning file from the Cisco IOS-XE image. Additional configuration is needed to configure the router to boot using the provisioning files and run using the individual modules.

When this command is used, copies of each module and the provisioning file within the Cisco IOS-XE image are copied and placed on the destination directory. The Cisco IOS-XE image file is unchanged after the operation is complete.

If the **to**destination-URL option is not entered, the Cisco IOS-XE image contents will be extracted onto the same directory where the Cisco IOS-XE image is currently stored.

If this command is used to extract individual module files onto a directory that already contains individual module files, the files that would have been extracted onto the same directory are instead extracted to an automatically created directory on the destination device.

Examples

The following example shows how to extract the individual modules and the provisioning file from a Cisco IOS-XE image that has already been placed in the directory where the user wants to store the individual modules and the provisioning file.

Output of the directory before and after the extraction is given to confirm the files were extracted.

```
Router# dir bootflash:
Directory of bootflash:/
   11  drwx       16384   Dec 4 2007 11:26:07 +00:00  lost+found
14401  drwx        4096   Dec 4 2007 11:27:41 +00:00  .installer
   12  -rw-   218783948   Dec 4 2007 12:12:16 +00:00
ASR1000rp1-advipservicesk9.01.00.00.12-33.XN.bin
Router# request platform software package expand file
bootflash:ASR1000rp1-advipservicesk9.01.00.00.12-33.XN.bin
Verifying parameters
Validating package type
Copying package files
Router# dir bootflash:
Directory of bootflash:/
   11  drwx       16384   Dec 4 2007 11:26:07 +00:00  lost+found
14401  drwx        4096   Dec 4 2007 11:27:41 +00:00  .installer
   12  -rw-   218783948   Dec 4 2007 12:12:16 +00:00
ASR1000rp1-advipservicesk9.01.00.00.12-33.XN.bin
28803  -rw-    52072652   Dec 4 2007 12:14:17 +00:00
asr1000rp1-espbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
28804  -rw-    21844172   Dec 4 2007 12:14:17 +00:00
asr1000rp1-rpaccess-k9.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
28805  -rw-    21520588   Dec 4 2007 12:14:18 +00:00
asr1000rp1-rpbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
28806  -rw-    24965324   Dec 4 2007 12:14:19 +00:00
asr1000rp1-rpcontrol.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
28807  -rw-    48515276   Dec 4 2007 12:14:20 +00:00
asr1000rp1-rpios-advipservicesk9.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
28808  -rw-    36954316   Dec 4 2007 12:14:21 +00:00
asr1000rp1-sipbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
28809  -rw-    19933388   Dec 4 2007 12:14:22 +00:00
asr1000rp1-sipspa.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
28802  -rw-        7145   Dec 4 2007 12:14:22 +00:00  packages.conf
928833536 bytes total (483700736 bytes free)
```

The following example shows how to extract the individual modules and the provisioning file from a Cisco IOS-XE image that has already been placed on the router in a directory that will not store the individual modules and the provisioning file. In this particular example, the contents of a Cisco IOS-XE image stored in usb0: are extracted into bootflash:.

Output of the bootflash: directory before and after the extraction is given to confirm the files were extracted.

```
Router# dir usb0:
Directory of usb0:/
1120  -rwx   213225676   Dec 4 2007 10:50:36 +00:00
asr1000rp1-advipservicesk9.v122_33_xn_asr_rls0_throttle.bin
Router# dir bootflash:

Directory of bootflash:/
   11  drwx       16384   Dec 4 2007 12:32:46 +00:00  lost+found
86401  drwx        4096   Dec 4 2007 14:06:24 +00:00  .ssh
14401  drwx        4096   Dec 4 2007 14:06:36 +00:00  .rollback_timer
43201  drwx        4096   Dec 4 2007 12:34:45 +00:00  .installer
Router# request platform software package expand file
usb0:asr1000rp1-advipservicesk9.v122_33_xn_asr_rls0_throttle.bin to bootflash:

Verifying parameters
Validating package type
Copying package files
Router# dir bootflash:
Directory of bootflash:/
   11  drwx       16384   Dec 4 2007 12:32:46 +00:00  lost+found
86401  drwx        4096   Dec 4 2007 14:06:24 +00:00  .ssh
14401  drwx        4096   Dec 4 2007 14:06:36 +00:00  .rollback_timer
43201  drwx        4096   Dec 4 2007 12:34:45 +00:00  .installer
28803  -rw-    51986636   Dec 4 2007 16:40:38 +00:00
asr1000rp1-espbase.v122_33_xn_asr_rls0_throttle.pkg
28804  -rw-    21838028   Dec 4 2007 16:40:39 +00:00
asr1000rp1-rpaccess-k9.v122_33_xn_asr_rls0_throttle.pkg
28805  -rw-    21508300   Dec 4 2007 16:40:39 +00:00
asr1000rp1-rpbase.v122_33_xn_asr_rls0_throttle.pkg
28806  -rw-    24963276   Dec 4 2007 16:40:40 +00:00
asr1000rp1-rpcontrol.v122_33_xn_asr_rls0_throttle.pkg
28807  -rw-    48419020   Dec 4 2007 16:40:41 +00:00
asr1000rp1-rpios-advipservicesk9.v122_33_xn_asr_rls0_throttle.pkg
28808  -rw-    36946124   Dec 4 2007 16:40:43 +00:00
asr1000rp1-sipbase.v122_33_xn_asr_rls0_throttle.pkg
28809  -rw-    14670028   Dec 4 2007 16:40:43 +00:00
asr1000rp1-sipspa.v122_33_xn_asr_rls0_throttle.pkg
28802  -rw-        6563   Dec 4 2007 16:40:43 +00:00  packages.conf
928862208 bytes total (708186112 bytes free)
```

Related Commands

Command	Description
request platform software package install file	Upgrades an individual module or a Cisco IOS-XE file.

request platform software package install commit

To cancel the rollback timer and commit a software upgrade, use the **request platform software package install commit**command in privileged EXEC or diagnostic mode.

request platform software package install rp *rp-slot-number* **commit** [**verbose**]

Syntax Description

rp rp-slot-number	Specifies the RP slot number.
commit	Specifies that an upgrade that was done using a rollback timer that has not expired can be committed.
verbose	(Optional) Displays verbose information, meaning all information that can be displayed on the console during the process will be displayed.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC (#)

Diagnostic Mode (diag)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.

Usage Guidelines

This command is entered after the **request platform software package install rp** *rp-slot-number* **file** command is used with the **auto-rollback** *minutes* option to begin an individual sub-package or a consolidated package upgrade. When the **auto-rollback** *minutes* option is used in this context, a rollback timer that cancels the upgrade after the number of specified *minutes*cancels the upgrade if the **request platform software package install rp** *rp-slot-number* **commit** command is not entered to commit the upgrade.

If this command is not entered after the **request platform software package install rp** *rp-slot-number* **file** command is used with the **auto-rollback** *minutes* option to upgrade an individual sub-package or a consolidated package and the rollback timer expires, the upgrade does not complete and the router continues running the previous sub-package or consolidated package.

Examples

In the following example, this command is entered to commit an upgrade:

request platform software package install rp 1 commit

Related Commands

Command	Description
request platform software package install file	Upgrades a consolidated package or sub-package.
request platform software package install rollback	Rolls back a previous software upgrade.

request platform software package install file

To upgrade a consolidated package or an individual sub-package, use the **request platform software package install file** command in privileged EXEC or diagnostic mode.

request platform software package install rp *rp-slot-number* **file** *file-URL* [**auto-rollback** *minutes*] [**provisioning-file** *URL*] [**slot** *slot-number*] [**bay** *bay-number*] [**force**] [**on-reboot**] [**verbose**]

Syntax Description

rp rp-slot-number	Specifies the route processor (RP) slot number.
file *file-URL*	Specifies the URL to the consolidated package or sub-package.
auto-rollback *minutes*	Specifies the setting of a rollback timer, and sets the number of minutes on the rollback timer before the rollback timer expires.
provisioning-file *provisioning-file-URL*	Specifies the URL to the provisioning file. A provisioning file is used for booting only when a Cisco ASR 1000 Series Aggregation Services Device is booted using individual sub-packages.
slot *slot-number*	Specifies the device slot number where a shared port adapter interface processor (SIP) can be installed.
bay *bay-number*	Specifies the shared port adapter (SPA) bay number within a SIP.
force	Specifies that the operation will be forced, meaning that the upgrade will proceed despite any warning messages.
on-reboot	Specifies that the installation will not be completed until the next RP reboot.
verbose	Displays verbose information, meaning all output that can be displayed on the console during the process will be displayed.

Command Default

If you do not enter the **request platform software package install file** command, the consolidated or sub package upgrades are not initiated on the device.

Command Modes

Privileged EXEC (#)

Diagnostic (diag)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.
Cisco IOS XE Release 3.8S	This command was modified. The **mdr** keyword was added.

Usage Guidelines

This command is used to upgrade consolidated packages and individual sub-packages.

When this command is used to upgrade a SIPBASE sub-package, the **slot** *slot-number* of the SIP must be specified.

When this command is used to upgrade a SIPSPA sub-package, the **slot** *slot-number*of the SIP and the **bay** *bay-number* of the SPA must be specified.

When the **auto-rollback** *minutes* option is used, the **request platform software package install rp** *rp-slot-number* **commit** command must be entered before the rollback timer expires to complete the upgrade. If this command is not entered, the device rolls back to the previous software version. The rollback timer expires after the number of specified *minutes*. If the **auto-rollback** *minutes* option is not used, the upgrade simply occurs.

Managing and Configuring a consolidated package using the request platform package command

In the following example, the **request platform software package install** command is used to upgrade a consolidated package running on RP 0. The **force** option, which forces the upgrade past any prompt (such as already having the same consolidated package installed), is used in this example.

```
Device# request platform software package install rp 0 file
bootflash:ASR1000rp1-advipservicesk9.01.00.00.12-33.XN.bin force

--- Starting installation state synchronization ---
Finished installation state synchronization
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Verifying image file locations
Locating image files and validating name syntax
Inspecting image file types
Processing image file constraints
Extracting super package content
Verifying parameters
Validating package type
Copying package files
Checking and verifying packages contained in super package
Creating candidate provisioning file

  WARNING:
  WARNING: Candidate software will be installed upon reboot
  WARNING:

Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
Determining whether installation is valid
Determining whether installation is valid ... skipped
Checking IPC compatibility with running software
Checking IPC compatibility with running software ... skipped
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Finished compatibility testing
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
SUCCESS: Software provisioned.  New software will load on reboot.
```

Device# **reload**

Note A reload must be performed to finish this procedure.

SIP Sub-package Installation with Verbose Option

In the following example, the SIP sub-package for the SIP in slot 1 is installed using the **request platform software package install**command. In this example, the **force** option, which forces the upgrade past any prompt (such as already having the same sub-package installed), and the **verbose** option, which displays all possible output during the installation, are used.

```
Device# request platform software package install rp 0
file bootflash:asr1000rp1-sipspa.v122_33_xn_asr_rls0_throttle.pkg slot 1 force verbose

--- Starting installation state synchronization ---
Finished installation state synchronization
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
... file names checked
Verifying image file locations
... image file locations verified
Locating image files and validating name syntax
... image file names validated
Inspecting image file types
... image file types acceptable
Processing image file constraints
... constraints satisfied
Creating candidate provisioning file
... created candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
... verified existing software set is valid
Processing candidate provisioning file
... candidate provisioning file processed
Constructing working set for candidate package set
... working set constructed
Constructing working set for running package set
... working set for running package set constructed
Checking command output
... command output is consistent with command set
Constructing merge of running and candidate packages
... merged running and candidate packages
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
... candidate package set is valid
Determining whether installation is valid
Software is unchanged
Software sets are identified as compatible
... installation is valid
Checking IPC compatibility with running software
calling minime_merge.sh for /tmp/tdlresolve/compat/_tmp_issu_provision_sw_
minime_merge done for /tmp/tdlresolve/compat/_tmp_issu_provision_sw_
```

```
... IPC is compatible with running software
Checking candidate package set infrastructure compatibility
... candidate package set infrastructure is compatible
Checking infrastructure compatibility with running software
... infrastructure is compatible with running software
Finished compatibility testing
--- Starting impact testing ---
Checking operational impact of change
... operational impact of change is allowable
Finished impact testing
--- Starting commit of software changes ---
Updating provisioning rollback files
... rollback provisioning files updated
Creating pending provisioning file
  Ensuring that cached content is written to media
... cached content flushed to media
... pending provisioning file created
Committing provisioning file
  Ensuring that cached content is written to media
  ... cached content flushed to media
... running provisioning file committed
Finished commit of software changes
--- Starting analysis of software changes ---
------------ changes to running software ------------
0 0 cc
----------------------------------------------------
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
... peer synchronization blocked
Creating the command set placeholder directory
  Finding latest command set
  ... latest command set identified
  Assembling CLI output libraries
  ... CLI output libraries assembled
  Assembling CLI input libraries
  ... CLI input libraries assembled
  Applying interim IPC and database definitions
    interim IPC and database definitions applied
      Replacing running software
      ... running software replaced
      Replacing CLI software
      ... CLI software replaced
      Restarting software
Restarting CC0
Restarting CC0
      ... software restarted
      Applying interim IPC and database definitions
*Oct  9 09:52:25.333: %MCP_OIR-6-OFFLINECARD: Card (cc) offline in slot 0
*Oct  9 09:52:25.334: %MCP_OIR-6-REMSPA: SPA removed from subslot 0/0,
interfaces disabled
*Oct  9 09:52:25.334: %MCP_OIR-6-REMSPA: SPA removed from subslot 0/1,
interfaces disabled
*Oct  9 09:52:25.334: %MCP_OIR-6-REMSPA: SPA removed from subslot 0/2,
interfaces disabled
*Oct  9 09:52:25.334: %MCP_OIR-6-REMSPA: SPA removed from subslot 0/3,
interfaces disabled      ... interim IPC and database definitions applied
      Notifying running software of updates
      ... running software notified
      Unblocking peer synchronization of operating information
      ... peer synchronization unblocked
    ... unmount of old packages scheduled
    Unmounting old packages
    ... inactive old packages unmounted
```

```
    Cleaning temporary installation files
    ... temporary installation files cleaned
    Finished update running software

SUCCESS: Finished installing software.
Device#
```

Upgrading SIP Sub-package without using the verbose option

In the following example, the SIP sub-package for the SIP in slot 1 is installed using the **request platform software package install**command. In this example, the **force** option, which forces the upgrade past any prompt (such as already having the same sub-package installed), is used. The **verbose** option is not used in this example.

```
Device# request platform software package install rp 0 file
bootflash:asr1000rp1-sipspa.v122_33_xn_asr_rls0_throttle.pkg slot 1 force

--- Starting installation state synchronization ---
Finished installation state synchronization
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Verifying image file locations
Locating image files and validating name syntax
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
Determining whether installation is valid
Software sets are identified as compatible
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Finished compatibility testing
--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
```

```
  Finding latest command set
  Assembling CLI output libraries
  Assembling CLI input libraries
  Applying interim IPC and database definitions
    interim IPC and database definitions applied
      Replacing running software
      Replacing CLI software
      Restarting software
Restarting CC1
Restarting CC1
      Applying interim IPC and database definitions
*Oct  9 09:54:55.365: %MCP_OIR-6-OFFLINECARD: Card (cc) offline in slot 1
*Oct  9 09:54:55.365: %MCP_OIR-6-REMSPA: SPA removed from subslot 1/1,
interfaces disabled
*Oct  9 09:54:55.365: %MCP_OIR-6-REMSPA: SPA removed from subslot 1/2,
interfaces disabled      Notifying running software of updates
      Unblocking peer synchronization of operating information
    Unmounting old packages
    Cleaning temporary installation files
      Finished update running software

SUCCESS: Finished installing software.
Device#
```

Upgrading IOS Sub-package

In the following example, the **request platform software package install**command is used to upgrade an IOS sub-package. In this example, the **force** option, which forces the upgrade past any prompt (such as already having the same module installed), is used.

```
Device# request platform software package install rp 0 file
bootflash:asr1000rp1-rpios-advipservicesk9.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
 force

--- Starting installation state synchronization ---
Finished installation state synchronization
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Verifying image file locations
Locating image files and validating name syntax
Inspecting image file types
    WARNING: In-service installation of IOSD package
    WARNING: requires software redundancy on target RP
    WARNING: or on-reboot parameter
    WARNING: Automatically setting the on-reboot flag
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:
```

```
WARNING: Candidate software combination not found in compatibility database
WARNING:
Determining whether installation is valid
Determining whether installation is valid ... skipped
Checking IPC compatibility with running software
Checking IPC compatibility with running software ... skipped
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
Checking infrastructure compatibility with running software ... skipped
Finished compatibility testing
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
SUCCESS: Software provisioned.  New software will load on reboot.
Device#
```

Note that the new RPIOS sub-package will become active only after a reboot. Reboot the device to finish this procedure.

Upgrading SPA Sub-package

In the following example, the **request platform software package install**command is use to upgrade a SIPSPA sub-package for the SPA in bay 0 of device slot 1. In this example, the **force** option, which forces the upgrade past any prompt (such as already having the same module installed), is used.

```
Device# request platform software package install rp 0 file
bootflash:asr1000rp1-sipspa.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg slot 1 bay 0
force

--- Starting installation state synchronization ---
Finished installation state synchronization
--- Starting file path checking ---
Finished file path checking
--- Starting image file verification ---
Checking image file names
Verifying image file locations
Locating image files and validating name syntax
Inspecting image file types
Processing image file constraints
Creating candidate provisioning file
Finished image file verification
--- Starting candidate package set construction ---
Verifying existing software set
Processing candidate provisioning file
Constructing working set for candidate package set
Constructing working set for running package set
Checking command output
Constructing merge of running and candidate packages
Finished candidate package set construction
--- Starting compatibility testing ---
Determining whether candidate package set is compatible
WARNING:
WARNING: Candidate software combination not found in compatibility database
WARNING:
Determining whether installation is valid
Software sets are identified as compatible
Checking IPC compatibility with running software
Checking candidate package set infrastructure compatibility
Checking infrastructure compatibility with running software
```

```
Finished compatibility testing
--- Starting impact testing ---
Checking operational impact of change
Finished impact testing
--- Starting commit of software changes ---
Updating provisioning rollback files
Creating pending provisioning file
Committing provisioning file
Finished commit of software changes
--- Starting analysis of software changes ---
Finished analysis of software changes
--- Starting update running software ---
Blocking peer synchronization of operating information
Creating the command set placeholder directory
  Finding latest command set
  Assembling CLI output libraries
  Assembling CLI input libraries
  Applying interim IPC and database definitions
    interim IPC and database definitions applied
      Replacing running software
      Replacing CLI software
      Restarting software
Restarting SPA CC1/0
      Applying interim IPC and database definitions
      Notifying running software of updates
      Unblocking peer synchronization of operating information
    Unmounting old packages
    Cleaning temporary installation files
      Finished update running software

SUCCESS: Finished installing software.
Device#
```

Related Commands

Command	Description
request platform software package install commit	Cancels the rollback timer and commits a software upgrade.
request platform software package install rollback	Rolls back a previous software upgrade.
request platform software package install snapshot	Creates a snapshot directory that will contain all the files extracted from a consolidated package.

request platform software package install rollback

To roll back a previous software upgrade, use the **request platform software package install rollback**command in privileged EXEC or diagnostic mode.

request platform software package install rp *rp-slot-number* **rollback** [{**as-booted** | **provisioning-file** *provisioning-file-URL*}] [**force**] [**on-reboot**] [**verbose**]

Syntax Description

rp rp-slot-number	Specifies the slot number of the RP doing the request.
as-booted	Specifies that the software update will not occur, and that the router will instead boot using the same procedure that it used during the last bootup.

provisioning-file *provisioning-file-URL*	Specifies that the software update will not occur, and that the router will instead boot using the specified provisioning file.
force	Specifies that the operation will be forced, meaning that the upgrade will proceed despite any warning messages.
on-reboot	Specifies that the installation will not be completed until the next RP reboot.
verbose	Displays verbose information, meaning all output that can be displayed on the console during the process will be displayed.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#)

Diagnostic (diag)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.

Usage Guidelines

This command rolls back a configuration that has an active rollback timer. Active rollback timers are used when the **auto-rollback** option is entered when software is being upgraded using the **request platform software package install file**command.

Examples

In the following example, an upgrade that was using a rollback timer is rolled back to the previous configuration instead of upgraded:

```
request platform software package install rp 0 rollback
```

Related Commands

Command	Description
request platform software package install commit	Cancel the rollback timer and commits a software upgrade.
request platform software package install file	Upgrades a consolidated package or an individual sub-package.

request platform software package install snapshot

To create a snapshot directory that contains all the files extracted from a consolidated package, use the **request platform software package install snapshot**command in privileged EXEC or diagnostic mode.

request platform software package install rp *rp-slot-number* **snapshot to** *URL* [**as** *snapshot-provisioning-filename*] [**force**] [**verbose**] [**wipe**]

Syntax Description

rp rp-slot-number	Specifies the slot number.

snapshot to *URL*	Creates a directory and extracts all files from the consolidated package into that directory. The directory is named in the command-line as part of the *URL_FS*. If the *URL_FS*is specified as a file system, the files in the consolidated package will be extracted onto the file system and not a directory on the file system.
as *snapshot-provisioning-filename*	(Optional) Renames the provisioning file in the snapshot directory. If this option is not used, the existing provisioning filename of the provisioning file in the consolidated package is used as the provisioning filename.
wipe	(Optional) Erases all content on the destination snapshot directory before extracting the files and placing them on the snapshot directory.
force	(Optional) Specifies that the operation will be forced, meaning that the upgrade will proceed despite any warning messages.
verbose	(Optional) Displays verbose information, meaning all output that can be displayed on the console during the process will be displayed.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#)

Diagnostic (diag)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.

Usage Guidelines

This command is used to create a directory at the destination device and extract the individual sub-packages in a consolidated package to that directory.

The **request platform software package expand** command is the only other command that can be used to extract individual sub-packages from a consolidated package.

Examples

In the following example, a snapshot directory named snapdir1_snap is created in the bootflash: file system, and the individual sub-package files from the consolidated package are extracted into the snapshot directory.

The second portion of the example first sets up the router to reboot using the files in the snapshot directory (deletes all previous boot system commands, configures the configuration register, then enters a boot system command to boot using the extracted provisioning file), saves the new configuration, then reboots so the router will boot using the extracted provisioning file, which allows the router to run using the extracted individual sub-package files.

```
Router(diag)# request platform software package install rp 0 snapshot to
bootflash:snapdir1_snap
--- Starting active image file snapshot --- Validating snapshot parameters Creating
destination directory Copying files to destination media
```

```
   Copied provisioning file as packages.conf
   Copying package file asr1000rp1-rpbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Copying package file asr1000rp1-rpcontrol.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg

   Copying package file
asr1000rp1-rpios-advipservicesk9.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
   Copying package file
asr1000rp1-rpaccess-k9.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Copying package file asr1000rp1-sipbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg

   Copying package file asr1000rp1-sipspa.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg
  Copying package file asr1000rp1-espbase.v122_33_xn_asr_rls0_throttle_20071204_051318.pkg

Moving files into final location Finished active image file snapshot
Router(config)# no boot system
Router(config)# config-register 0x1
Router(config)# boot system harddisk:snapdir1_snap/packages.conf
Router(config)# exit
*May 11 01:31:04.815: %SYS-5-CONFIG_I: Configured from console by con
Router# write mem
Building configuration...
[OK]
```

Router# **reload**

Related Commands

Command	Description
request platform software package install file	Upgrades a consolidated package or an indivual sub-package.

request platform software process release

To restart processes that have been placed in the hold down state by the Process Manager on the Cisco ASR 1000 Series Routers, use the **request platform software process release**command in privileged EXEC or diagnostic mode.

request platform software process release *slot* **all**

Syntax Description

slot	Specifies the hardware slot. Options include: • *number* --The number of the SIP slot of the hardware module where the trace level is being set. For instance, if you wanted to specify the SIP in SIP slot 2 of the router, enter 2 as the *number*. • **f0** --The ESP in ESP slot 0. • f1 --The ESP in ESP slot 1 • **fp active** --The active ESP. • **fp standby** --The standby ESP. • **r0** --The RP in RP slot 0. • r1 --The RP in RP slot 1. • **rp active** --The active RP. • **rp standby** --The standby RP.
all	Specifies that all processes currently in the holddown state within the selected slot will be restarted.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#) Diagnostic Mode (diag)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.

Usage Guidelines

This command is used to restart processes in the holddown state. If a process is in the holddown state, a console message is generated to notify the user that the process is helddown.

Before placing any process in the holddown state, the Process Manager makes up to 5 attempts over 120 seconds to enable the process. These attempts to enable the process also happen automatically at startup. If the Process Manager is unable to enable the process after these attempts, the process will then be placed in the holddown state.

When this command is entered, it only attempts to restart processes currently in the holddown state. Active processes will not be affected by entering this command.

Examples

In the following example, this command is entered to restart any process currently on RP 0 in the holddown state:

```
request platform software process release r0 all
```

request platform software system shell

To request platform shell access, use the **request platform software system shell**command in privileged EXEC mode.

request platform software system shell [{**rp** | **esp** | **sip**}]

Syntax Description

rp	Specifies the Route Processor (RP); it can be either active or standby.
esp	Specifies the Embedded Services Processor (ESP) control processor; it can be either active or standby.
sip	Specifies the SPA Interface Processor (SIP).

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(33)XNC	This command was introduced.

Usage Guidelines

The platform shell command needs to be entered before before using the request platform software system shell command. Providing shell access would not be necessary. However, there might be some cases where the command may not be available, or the IOS process hangs, or IOS console may not be available. In such cases, you can login to the shell and see the status of the system.

The shell should be accessed under Cisco supervision, and no support is provided if accessed without supervision. The following message is displayed , before the shell access is granted:

"Activity within this shell can jeopardize the functioning of the system.

Use this functionality only under supervision of Cisco Support."

Examples

In the following example,

```
Router(config)# platform shell
Router(config)# exit
Router# request platform software shell system
Activity within this shell can jeopardize the functioning of the system.
Are you sure you want to continue? [y/n] y
**********************************************************************
Activity within this shell can jeopardize the functioning
of the system.
Use this functionality only under supervision of Cisco Support.
```

Related Commands

Command	Description
platform shell	Grants shell and enters shell access grant configuration mode.

request platform software shell session output format

To modify the format of the output of some **show** commands on the Cisco ASR1000 Series Routers, use the **request platform software shell session output format**command in privileged EXEC and diagnostic mode.

request platform software shell session output format *format*

Syntax Description

format	Specifies the output format for **show** command output. Options include: • **html** --Specifies Hypertext Markup Language (HTML) output. • **raw** --Specifies the raw message output. • **text** --Specifies plaintext output, which is the default. • **xml** -- Specifies Extensible Markup Language (XML) output

Command Default

All **show** command output is seen in plaintext (the **text** *format)*by default.

Command Modes

Privileged EXEC (#) Diagnostic Mode (diag)

Command History

Release	Modification
IOS XE Release 2.1	This command was introduced

Usage Guidelines

Entering this command can only change the output of some **show** commands that are available in both privileged EXEC and diagnostic mode. At the current time, most of these commands are **show platform software** and **show platform hardware** commands.

Only a small subset of commands currently produce output using the **html** option.

Examples

In the following example, the **request platform software shell session output format** command is used to change the show output format from **text** to **raw**. The output of the **show platform hardware slot r0 alarms visual** command is shown both before and after the **request platform software shell session output format** command was entered to illustrate the change in output format.

```
Router# show platform hardware slot r0 alarms visual

Current Visual Alarm States
Critical: On
Major   : On
Minor   : Off
Router# request platform software shell session output format raw
Router# show platform hardware slot r0 alarms visual

message@alarms_msg: {
  tdl_cman_alarms_data@tdl_cman_alarms_data: {
    critical@tdl_boolean:TDL_TRUE
    major@tdl_boolean:TDL_TRUE
    minor@tdl_boolean:TDL_FALSE
  }
}
message@ui_req_msg: {
  ui_req@ui_req: {
    request_id@U64:2
    client@ui_client: {
      location@svc_loc: {
        fru@b_fru:BINOS_FRU_RP
        slotnum@I16:0
        baynum@I16:0
      }
      client_type@ui_client_type:UICLIENT_INVALID
      term_type@ui_terminal_type:UITT_INVALID
```

```
      ttynum@U32:0
      tty_name@NS:
      user_name@NS:
    }
    command@NS:
    request_name@NS:
    flags@ui_req_flag:
  }
}
```

In the following example, the **request platform software shell session output format** command is used to change the show output format from **text** to **xml**. The output of the **show platform hardware slot r0 alarms visual** command is shown both before and after the **request platform software shell session output format** command was entered to illustrate the change in output format.

```
Router# show platform hardware slot r0 alarms visual

Current Visual Alarm States
Critical: On
Major   : On
Minor   : Off
Router# request platform software shell session output format xml
Router# show platform hardware slot r0 alarms visual

<?xml version="1.0"?>
<iossr-response action="3">
<cmd-response>
<alarms_msg><tdl_cman_alarms_data><critical><TDL_TRUE/></critical>
<major><TDL_TRUE/></major>
<minor><TDL_FALSE/></minor>
</tdl_cman_alarms_data>
</alarms_msg>
<ui_req_msg><ui_req><request_id>4</request_id>
<client><location><fru><BINOS_FRU_RP/></fru>
<slotnum>0</slotnum>
<baynum>0</baynum>
</location>
<client_type><UICLIENT_INVALID/></client_type>
<term_type><UITT_INVALID/></term_type>
<ttynum>0</ttynum>
<tty_name></tty_name>
<user_name></user_name>
</client>
<command></command>
<request_name></request_name>
<flags></flags>
</ui_req>
</ui_req_msg>
</cmd-response>
</iossr-response>
```

request platform software snapshot

To take a snapshot of the bootflash, use the **request platform software snapshot** command in privilege EXEC mode.

request platform software snapshot *slot* {**cancel** | **create** | **delete** | **restore**}*name*

Syntax Description

snapshot	Requests snapshot actions.

slot	Specifies the hardware slot. Options include: • *number* --The number of the SIP slot of the hardware module where the trace level is being set. For instance, if you wanted to specify the SIP in SIP slot 2 of the router, enter 2 as the *number*. • **f0** --The ESP in ESP slot 0. • f1 --The ESP in ESP slot 1 • **fp active** --The active ESP. • **fp standby** --The standby ESP. • **r0** --The RP in RP slot 0. • r1 --The RP in RP slot 1. • **rp active** --The active RP. • **rp standby** --The standby RP.
cancel	Cancels a snapshot operation.
create	Creates a snapshot
delete	Deletes a snapshot
restore	Restores a snapshot
name	Specifies the name of the snapshot to be modified.

Command Default

No default behavior or values

Command Modes

Privileged EXEC (#) Diagnostic Mode (diag)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.

Usage Guidelines

Use the **request platform software snapshot** command to create a snapshot of the bootflash, including the NVRAM partitions and the ROMMON memory, on the harddisk. This command can also be used to restore a snapshot.

Examples

This example shows how to create a snapshot named "stan" on the processor in the RO slot.

```
router#request platform software snapshot R0 create stan
```

Related Commands

Command	Description
show platform software snapshot status	Use this command to display a snapshot of the bootflash.

request platform software vty attach

To enter EXEC mode on a router after persistent SSH or persistent Telnet is configured to connect to the router in diagnostic mode, use the **request platform software vty attach**command in diagnostic mode.

request platform software vty attach [**permanent**]

Syntax Description

permanent	(Optional) Specifies that the router should not return to diagnostic mode if EXEC mode is exited.

Command Default

No default behavior or values

Command Modes

Diagnostic (diag)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.

Usage Guidelines

If persistent Telnet or persistent SSH is configured to make users wait for an IOS vty line before allowing them to access the IOS CLI, this command can be used to attach to an IOS vty line and place the user in EXEC mode. Exiting EXEC mode returns the user to diagnostic mode unless the **permanent** keyword is entered. When the **permanent** keyword is entered, exiting EXEC mode exits the router.

The vty lines must be configured to allow local login for this command to work. The vty lines must also be configured to accept the type of transport traffic (SSH or Telnet) being used to connect to the router for the session in which the **request platform software vty attach** command is entered.

Examples

In the following example, this command is used to leave diagnostic mode and enter privileged EXEC mode:

```
Router(diag)#
request platform software vty attach
Router#
```

In the following example, this command is used to leave diagnostic mode and enter privileged EXEC mode. The user then re-enters diagnostic mode by exiting privileged EXEC mode:

```
Router(diag)# request platform software vty attach
Router# exit

Router(diag)#
```

In the following example, this command is used with the **permanent** option to leave diagnostic mode and enter privileged EXEC mode. The user then exits the router by exiting privileged EXEC mode:

```
Router(diag)# request platform software vty attach permanent
Router# exit
Connection to Router closed.
```

revision

To set the revision number for the Multiple Spanning Tree (802.1s) (MST) configuration, use the **revision** command in MST configuration submode. To return to the default settings, use the **no** form of this command.

revision *version*
no revision

Syntax Description

version	Revision number for the configuration; valid values are from 0 to 65535.

Command Default

version is **0**

Command Modes

MST configuration (config-mst)

Command History

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

Usage Guidelines

Two Cisco 7600 series routers that have the same configuration but different revision numbers are considered to be part of two different regions.

Caution Be careful when using the **revision** command to set the revision number of the MST configuration because a mistake can put the switch in a different region.

Examples

This example shows how to set the revision number of the MST configuration:

```
Router(config-mst)# revision 5
Router(config-mst)#
```

Related Commands

Command	Description
instance	Maps a VLAN or a set of VLANs to an MST instance.
name (MST configuration submode)	Sets the name of an MST region.
show	Verifies the MST configuration.
show spanning-tree	Displays information about the spanning-tree state.
spanning-tree mst configuration	Enters MST-configuration submode.

rmdir

To remove an existing directory in a Class C Flash file system, use the **rmdir** command in EXEC, privileged EXEC, or diagnostic mode.

rmdir *directory*

Syntax Description

directory	Directory to delete.

Command Modes

User EXEC

Privileged EXEC

Diagnostic

Command History

Release	Modification
11.3 AA	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE Release 2.1	This command was introduced on the Cisco ASR1000 Series Router and was made available in diagnostic mode.

Usage Guidelines

This command is valid only on Class C Flash file systems.

Caution You can use the **rmdir** command to remove a directory that another user is currently accessing in read-only mode, for example if it is that user's default working directory. If you use the **rmdir** command to remove such a directory and a user whose current directory is set to the deleted directory then uses the **pwd** command to display the current working directory, the following error message is displayed: Cannot determine current directory.

Examples

The following example deletes the directory named newdir:

```
Router# dir

Directory of flash:
  2  drwx           0   Mar 13 1993 13:16:21  newdir
8128000 bytes total (8126976 bytes free)
Router# rmdir newdir

Rmdir file name [newdir]?
Delete flash:newdir? [confirm]
Removed dir flash:newdir
Router# dir

Directory of flash:
No files in directory
8128000 bytes total (8126976 bytes free)
```

Related Commands

Command	Description
dir	Displays a list of files on a file system.
mkdir	Creates a new directory in a Class C Flash file system.

rommon-pref

To select a ReadOnly or Upgrade ROMmon image to be booted on the next reload of a Cisco 7200 VXR router or Cisco 7301 router when you are in ROMmon, use the **rommon-pref**command in ROM monitor mode.

rommon-pref [{**readonly** | **upgrade**}]

Syntax Description

readonly	Selects the ReadOnly ROMmon image to be booted on the next reload.
upgrade	Selects the Upgrade, second ROMmon image to be booted on the next reload.

Command Default

No default behavior or values

Command Modes

ROM monitor mode

Command History

Release	Modification
12.0(28)S	This command was introduced on the Cisco 7200 VXR router. It was introduced in ROMmon version 12.3(4r)T1 for the Cisco 7200 VXR router.
12.3(8)T	This command was integrated into Cisco IOS Release 12.3(8)T and supported on the Cisco 7200 VXR router and Cisco 7301 router. It was introduced in ROMmon version 12.3(4r)T2 for the Cisco 7301 router.
12.3(9)	This command was integrated into Cisco IOS Release 12.3(9) and supported on the Cisco 7200 VXR router and Cisco 7301 router.

Usage Guidelines

You might select the ReadOnly ROMmon image to be booted on the next reload because the Upgrade image has features or side effects you do not like.

When you are in ROMmon, there is no descriptive output to inform you whether the ReadOnly ROMmon image was reloaded. To confirm the reload, use the showmon command after entering the rommon-pref readonly command.

Use this command when you are in ROMmon mode. Use the **upgrade rom-monitor preference**command when you are in Cisco IOS.

Examples

The following example, applicable to both the Cisco 7200 VXR and Cisco 7301 routers, shows how to select the ReadOnly ROMmon image to be booted on the next reload of the router when you are already in ROMmon mode:

```
rommon 2 > rommon-pref readonly
```

Related Commands

Command	Description
showmon	Shows both the ReadOnly and the Upgrade ROMmon image versions when you are in ROMmon mode, as well as which ROMmon image is running.

route-converge-interval

To configure the time interval after which the old FIB entries are purged, use the **route-converge-interval**command in main CPU submode. To return to the default settings, use the **no** form of this command.

route-converge-interval *seconds*
no route-converge-interval

Syntax Description

seconds	Time interval, in seconds, after which the old FIB entries are purged ; valid values are from 60 to 3600 seconds.

Command Default

seconds is **120** seconds (2 minutes).

Command Modes

Main CPU submode

Command History

Release	Modification
12.2(17b)SXA	Support for this command was introduced on the Supervisor Engine 720.
12.2(18)SXD	This command is supported on releases prior to Release 12.2(18)SXD.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

SRM/SSO is supported in the following releases only.

- Release 12.2(17b)SXA and later rebuilds of Release 12.2(17b)SXA
- Release 12.2(17d)SXB and later rebuilds of Release 12.2(17d)SXB

This command is not supported in Cisco 7600 series routers that are configured with a Supervisor Engine 2.

The time interval for route-converge delay is needed to simulate the route-converge time when routing protocols restart on switchover.

Examples

This example shows how to set the time interval for the route-converge delay:

```
Router(config)# redundancy
Router(config-red)# main-cpu
Router(config-red-main)# route-converge-interval 90
Router(config-red-main)#
```

This example shows how to return to the default time interval for the route-converge delay:

```
Router(config)# redundancy
Router(config-red)# main-cpu
```

```
Router(config-red-main)# no route-converge-interval
Router(config-red-main)#
```

Related Commands

Command	Description
redundancy	Enters redundancy configuration mode.

rsh

To execute a command remotely on a remote shell protocol (rsh) host, use the **rsh**command in privilegedEXEC mode.

rsh {*ip-addresshost*} [**/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.
remote-command	Command to be executed remotely.

Command Default

If you do not specify the **/user** *username* keyword and argument, the Cisco IOS software sends a default remote username. As the default value of the remote username, the software sends the username associated with the current tty process, if that name is valid. For example, if the user is connected to the router through Telnet and the user was authenticated through the **username**command, then the software sends that username as the remote username. If the tty username is invalid, the software uses the host name as the both the remote and local usernames.

Note For Cisco, tty lines are commonly used for access services. The concept of tty originated with UNIX. For UNIX systems, each physical device is represented in the file system. Terminals are sometimes called tty devices (tty stands for teletype, the original UNIX terminal).

Command Modes

Privileged EXEC

Command History

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

Usage Guidelines

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

For security reasons, the software does not default to a remote login if no command is specified, as does UNIX. Instead, the router provides Telnet and connect services that you can use rather than rsh.

Examples

The following command specifies that the user named sharon attempts to remotely execute the UNIX **ls** command with the *-a* argument on the remote host named mysys.cisco.com. The command output resulting from the remote execution follows the command example:

```
Router1# rsh mysys.cisco.com /user sharon ls -a
.
.
.
.alias
.cshrc
.emacs
.exrc
.history
.login
.mailrc
.newsrc
.oldnewsrc
.rhosts
.twmrc
.xsession
jazz
```

scheduler allocate

To guarantee CPU time for processes, use the **scheduler allocate**command in global configuration mode. To restore the default, use the **no** form of this command.

scheduler allocate *interrupt-time process-time*
no scheduler allocate

Syntax Description

interrupt-time	Integer (in microseconds) that limits the maximum number of microseconds to spend on fast switching within any one network interrupt context. The range is from 400 to 60000 microseconds. The default is 4000 microseconds.
process-time	Integer (in microseconds) that guarantees the minimum number of microseconds to spend at the process level when network interrupts are disabled. The range is from 100 to 4000 microseconds.The default is 200 microseconds. The default for Catalyst 6500 series switches and Cisco 7600 series routers is 800 microseconds.

Command Default

Approximately 5 percent of the CPU is available for process tasks.

Command Modes

Global configuration

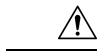
Command History

Release	Modification
11.2	This command was introduced.
12.2(14)SX	Support for this command was introduced on the Supervisor Engine 720.

Release	Modification
12.2(17a)SX	This command was changed as follows: • The *process-time* default setting was changed from 200 microseconds to 800 microseconds. • The **no scheduler allocate** action was changed to return to the default settings.
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was extended to the 12.2(17d)SXB release.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

This command applies to the Catalyst 6500 series switches, Cisco 7200 series, Cisco 7500 series, and Cisco 7600 series routers.



Caution We recommend that you do not change the default settings. Changing settings associated with CPU processes can negatively impact system performance.

Entering the **scheduler allocate** command without arguments is the same as entering the **no scheduler allocate** or the **default scheduler allocate** command.

Examples

The following example makes 20 percent of the CPU available for process tasks:

```
Router(config)# scheduler allocate 2000 500
```

Related Commands

Command	Description
scheduler interval	Controls the maximum amount of time that can elapse without running system processes.

scheduler heapcheck enable

To enable heapcheck processing, use the **scheduler heapcheck enable**command in global configuration mode. To disable scheduler heapcheck processing, use the **no** form of this command.

scheduler heapcheck enable
no scheduler heapcheck enable

Syntax Description

This command has no arguments or keywords.

Command Default

The **scheduler heapcheck enable**command is disabled by default. If no keywords are specified, scheduler heapcheck processing will not be performed.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.2(1)T	This command was introduced in the Cisco IOS Release 15.2(1)T.

Examples

The following example shows how to enable scheduler heapcheck processing:

```
Router# configure terminal
Router(config)# scheduler heapcheck enable
```

Related Commands

Command	Description
scheduler heapcheck process	Performs a sanity check for corruption in memory blocks when a process switch occurs.

scheduler heapcheck poll

To validate the memory and edisms poll routine, use the **scheduler heapcheck poll**command in global configuration mode. To disable the memory check and edisms poll routine, use the **no** form of this command.

scheduler heapcheck poll
no scheduler heapcheck poll

Syntax Description

This command has no arguments or keywords.

Command Default

The **scheduler heapcheck poll**command is disabled by default. If no keywords are specified, a sanity check is performed on all the memory blocks and memory pools.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.
12.2(33)SRC	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SRC.
12.2(33)SXI	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SXI.

Examples

The following example shows how to validate the memory check and edisms poll routine:

```
Router# configure terminal
Router(config)# scheduler heapcheck poll
```

Related Commands

Command	Description
scheduler heapcheck process	Performs a sanity check for corruption in memory blocks when a process switch occurs.

scheduler heapcheck process

To perform a "sanity check" for corruption in memory blocks when a process switch occurs, use the **scheduler heapcheck process**command in global configuration mode. To disable this feature, use the **no** form of this command.

scheduler heapcheck process [**memory** [**fast**] [**io**] [**multibus**] [**pci**] [**processor**] [**checktype** {**all** | **data** | **magic** | **mlite-data** | **pointer** | **refcount** | **lite-chunks**}]]
no scheduler heapcheck process

Syntax Description

memory	(Optional) Specifies checking all memory blocks and memory pools.
fast	(Optional) Specifies checking the fast memory block.
io	(Optional) Specifies checking the I/O memory block.
multibus	(Optional) Specifies checking the multibus memory block.
pci	(Optional) Specifies checking the process control information (PCI) memory block.
processor	(Optional) Specifies checking the processor memory block.
checktype	(Optional) Specifies checking specific memory pools.
all	(Optional) Specifies checking the value of the block magic, red zone, size, refcount, and pointers (next and previous).
data	(Optional) Specifies checking the value of normal blocks.
magic	(Optional) Specifies checking the value of the block magic, red zone, and size.
mlite-data	(Optional) Specifies checking the value of memory allocation lite (malloc-lite) blocks.
pointer	(Optional) Specifies checking the value of the next and previous pointers.
refcount	(Optional) Specifies checking the value of the block magic and refcount.
lite-chunks	(Optional) Specifies checking the memory blocks allocated by the memory allocation lite (malloc_lite) feature.

Command Default

This command is disabled by default. If no keywords are specified, a sanity check will be performed on all the memory blocks and memory pools.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(15)T	This command was introduced.
12.3(11)T	The **lite-chunks** keyword was added.
12.4(20)T	The **data**and **mlite-data**keywords were added.

Usage Guidelines

When configuring this command, you can choose none or all memory block keywords (**fast**, **io**, **multibus**, **pci**, **processor**, and **checktype**).

Enabling this command has a significant impact on router performance.

Examples

The following example shows how to sanity check for corruption in the I/O memory block when a process switch occurs. In this example, the values of only the block magic, red zone, and size will be checked.

```
scheduler heapcheck process memory io checktype magic
```

The following example shows how to sanity check for corruption in the processor memory block when a process switch occurs. In this example, the values of only the next and previous pointers will be checked.

```
scheduler heapcheck process memory processor checktype pointer
```

Related Commands

Command	Description
memory lite	Enables the malloc_lite feature.
memory sanity	Performs a "sanity check" for corruption in buffers and queues.

scheduler interrupt mask profile

To start interrupt mask profiling for all processes running on the system, use the **scheduler interrupt mask profile**command in global configuration mode. To stop interrupt mask profiling, use the **no** form of this command.

scheduler interrupt mask profile
no scheduler interrupt mask profile

Syntax Description

This command has no arguments or keywords.

Command Default

Interrupt mask profiling is disabled by default.

Command Modes

Global configuration

Command History

Release	Modification
12.4(2)T	This command was introduced.

Usage Guidelines

This command enables the collection of details regarding the total amount of time a process has masked interrupts since the interrupt mask profiler was enabled.

Examples

The following example shows how to enable interrupt mask profiling:

```
Router(config)# scheduler interrupt mask profile
```

Related Commands

Command	Description
clear processes interrupt mask detail	Clears the interrupt masked details for all processes and stack traces that have been dumped into the interrupt mask buffer.
scheduler interrupt mask size	Configures the maximum number of entries that can exist in the interrupt mask buffer.
scheduler interrupt mask time	Configures the maximum allowed time that a process can run with interrupts masked.
show process interrupt mask buffer	Displays the information stored in the interrupt mask buffer.
show processes interrupt mask detail	Displays interrupt masked details for the specified process or all processes in the system.

scheduler interrupt mask size

To configure the maximum number of entries that can exist in the interrupt mask buffer, use the **scheduler interrupt mask size**command in global configuration mode. To reset the maximum number of entries that can exist in the interrupt mask buffer to the default, use the no form of this command.

scheduler interrupt mask size *buffersize*
no scheduler interrupt mask size

Syntax Description

buffersize	Specifies the number of entries that can exist in the interrupt mask buffer.

Command Default

The default buffer size is 50 entries.

Command Modes

Global configuration

Command History

Release	Modification
12.4(2)T	This command was introduced.

Examples

The following example shows how to configure 100 entries the maximum number of entries that can exist in the interrupt mask buffer:

```
Router(config)# scheduler interrupt mask size 100
```

Related Commands

Command	Description
clear processes interrupt mask detail	Clears the interrupt masked details for all processes and stack traces that have been dumped into the interrupt mask buffer.
scheduler interrupt mask profile	Enables or disables interrupt mask profiling for all processes running on the system.

Command	Description
scheduler interrupt mask time	Configures the maximum amount of time a process can run with interrupts masked.
show processes interrupt mask buffer	Displays interrupt masked details for the specified process or all processes in the system and displays information stored in the interrupt mask buffer.
show processes interrupt mask detail	Displays interrupt masked details for the specified or all processes in the system.

scheduler interrupt mask time

To configure the maximum time that a process can run with interrupts masked before another entry is created in the interrupt mask buffer, use the **scheduler interrupt mask time**command in global configuration mode. To reset the threshold time to the default, use the **no** form of this command.

scheduler interrupt mask time *threshold-time*
no scheduler interrupt mask time

Syntax Description

threshold-time	Specifies the maximum amount of timein microseconds a process can be in interrupt masked state without creating an entry in the interrupt mask buffer.

Command Default

The default threshold time value is 50 microseconds.

Command Modes

Global configuration

Command History

Release	Modification
12.4(2)T	This command was introduced.

Examples

The following shows how to configure 100 microseconds as the maximum time a process can run with interrupts masked before another entry is created in the interrupt mask buffer:

```
Router(config)# scheduler interrupt mask time 100
```

Related Commands

Command	Description
clear processes interrupt mask detail	Clears the interrupt masked details for all processes and stack traces that have been dumped into the interrupt mask buffer.
scheduler interrupt mask profile	Enables or disables interrupt mask profiling for all processes running on the system.
scheduler interrupt mask size	Configures the maximum number of entries that can exist in the interrupt mask buffer.

Command	Description
show processes interrupt mask buffer	Displays the information stored in the interrupt mask buffer.
show processes interrupt mask detail	Displays interrupt masked details for the specified process or all processes in the system.

scheduler interval

To control the maximum amount of time that can elapse without running system processes, use the **scheduler interval** command in global configuration mode. To restore the default, use the **no**form of this command.

scheduler interval *milliseconds*
no scheduler interval

Syntax Description

milliseconds	Integer that specifies the interval (in milliseconds). The minimum interval that you can specify is 500 milliseconds; there is no maximum value.

Command Default

High-priority operations are allowed to use as much of the CPU as needed.

Command Modes

Global configuration

Command History

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

Usage Guidelines

The normal operation of the network server allows the switching operations to use as much of the central processor as is required. If the network is running unusually heavy loads that do not allow the processor the time to handle the routing protocols, give priority to the system process scheduler. High-priority operations are allowed to use as much of the CPU as needed.

Note Changing settings associated with CPU processes can negatively impact system performance.

On the Cisco 7200 series and Cisco 7500 series, use the **scheduler allocate** global configuration command instead of the **scheduler interval** command.

Examples

The following example changes the low-priority process schedule to an interval of 750 milliseconds:

```
Router(config)# scheduler interval 750
```

Related Commands

Command	Description
scheduler allocate	Guarantees CPU time for processes.

scheduler isr-watchdog

To detect if an Interrupt Service Routine (ISR) is suspended or stalled and to schedule and manage a watchdog timeout on an ISR, use the **scheduler isr-watchdog**command in global configuration mode. To disable the configuration, use the **no** form of this command.

scheduler isr-watchdog
no scheduler isr-watchdog

Syntax Description

There are no additional keywords or arguments with this command.

Command Default

The default detection time is 2 minutes.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS 15.0(1)M.

Usage Guidelines

The timer ISR checks the current context to avoid holding processes accountable for CPU time spent servicing interrupts during the process time slice, and vice versa for interrupt-level code accountability. However, at each timer tick, the timer ISR applies the full 4 milliseconds of CPU time to the current context. As a result, depending on when the timer tick occurs in relation to a context switch, you might see inaccuracies in CPU utilization accounting compared with the actual computation time because some or all of the tick is being applied to the wrong context.

Examples

The following example shows how to detect if an ISR is suspended or stalled and to manage a watchdog timeout on an ISR:

```
Router> enable
Router# configure terminal
Router(config)# scheduler isr-watchdog
```

Related Commands

Command	Description
scheduler max-sched-time	Configures the maximum time in milliseconds that a scheduler can run without flagging an error.

scheduler max-sched-time

To configure or change the maximum time, in milliseconds that a scheduler can run without flagging an error or overload of the CPU, use the **scheduler max-sched-time**command in global configuration mode. To disable this configuration, use the **no** form of this command.

scheduler max-sched-time *milliseconds*
no scheduler max-sched-time

Syntax Description

milliseconds	The maximum time, in milliseconds (ms). The range is from 1 to 3600.

Command Default The default time is 2000 ms to signal an overload of the CPU.

Command Modes Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS 15.0(1)M.

Usage Guidelines The default behavior of the **scheduler max-sched-time** command is to stop the process only if it is fatal. A task is defined as fatal if the task gets another watchdog within 12 hours of being assigned the first watchdog, and a handler has been registered.

Examples The following example shows how to configure the maximum time in milliseconds (to 1000 ms in this example) that a scheduler can run without flagging an error:

```
Router> enable
Router# configure terminal
Router(config)# scheduler max-sched-time 1000
```

Related Commands

Command	Description
scheduler isr-watchdog	Detects if an ISR is suspended or stalled and manages a watchdog timeout on an ISR.

scheduler process-watchdog

To configure the default action of a watchdog timeout for a process using a scheduler, use the **scheduler process-watchdog**command in global configuration mode. To disable the configuration, use the **no** form of this command.

scheduler process-watchdog {**hang** | **normal** | **reload** | **terminate**}
no scheduler process-watchdog

Syntax Description

hang	Retains the process but does not schedule it.
normal	Enables factory-specified per-process behavior.
reload	Reloads the system.
terminate	Terminates the process and continues.

Command Default The default value is **normal**.

Command Modes Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS 15.0(1)M.

Usage Guidelines

The watchdog timer sets the interval after which the scheduler assumes a process has been suspended or stalled and needs to be stopped.

Examples

The following example shows how to configure the default action of a watchdog timeout for a process using a scheduler:

```
Router> enable
Router# configure terminal
Router(config)# scheduler process-watchdog normal
```

Related Commands

Command	Description
scheduler max-sched-time	Configures the maximum time in milliseconds that a scheduler can run without flagging an error.

scheduler timercheck process

To configure process-level timer validation on a scheduler, and check the timer tree of the process after every context switch of the process Packet Identification number (PID) is configured, use the **scheduler timercheck process**command in global configuration mode. To disable this configuration, use the **no** form of this command.

scheduler timercheck process *pid*
no scheduler timercheck process *pid*

Syntax Description

pid	PID number in the range is from 1 to 2147483647.

Command Default

The process-level timer validation is not configured on a scheduler.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS 15.0(1)M.

Usage Guidelines

Specify the **show processes timercheck** command after configuring the **schedule timercheck process** command to display the details of the configuration.

Examples

The following example shows how to configure process-level timer validation on a scheduler with a PID value of 5:

```
Router> enable
Router# configure terminal
Router(config)# scheduler timercheck process 5
Router# show processes timer
System timer check not configured.
Process timer check configuration follows.
PID   Configuration            Name
1     On every context switch. Chunk Manager
```

Related Commands

Command	Description
show processes timercheck	Displays information about the active Cisco IOS processes or the Cisco IOS Software Modularity POSIX-style processes.
scheduler timercheck system context	Configures system-level validation on context switches on a scheduler.

scheduler timercheck system context

To configure system-level validation on context switches on a scheduler, and check system level-timers, use the **scheduler timercheck system context**command in global configuration mode. To disable the configuration, use the **no** form of this command.

scheduler timercheck system context
no scheduler timercheck system context

Syntax Description

This command has no additional keywords or arguments.

Command Default

The system-level validation on context switches on a scheduler is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS 15.0(1)M.

Examples

The following example shows how to configure system level validation on context switches on a scheduler:

```
Router> enable
Router# configure terminal
Router(config)# scheduler timercheck system context
```

Related Commands

Command	Description
scheduler timercheck process	Configures process-level timer validation on a scheduler.

send

To send messages to one or all terminal lines, use the **send**command in user or privileged EXEC mode.

send {**line-number** | ***** | **aux** *number* | **console** *number* | **log** *number* [*msg-ext*] | **tty** *number* | **vty** *number* | **xsm** [**client** *client-id*] **message** *text*}

Syntax Description

line-number	Line number to which the message will be sent.

*****	Sends a message to all lines.
aux *number*	Sends a message to the specified auxiliary (AUX) port.
console *number*	Sends a message to the specified console port.
log *number*	Logs a message of the specified severity.
msg-text	Logging message text.
client *client-id*	(Optional) Sends the message to the specified client. The message is sent to all clients if the client ID is not specified.
message *text*	Sends a message to XSM client when it is used with the **xsm** keyword.
tty *number*	Sends a message to the specified asynchronous line.
vty *number*	Sends a message to the specified virtual asynchronous line.
xsm *client-id*	Sends a message to the XML Subscription Manager (XSM) client.

Command Default

No messages are sent.

Command Modes

User EXEC (>)

Priviledged EXEC (#)

Command History

Release	Modification
11.2	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.
15.0(1)M	This command was integrated into Cisco IOS Release 15.0(1)M.
Cisco IOS XE Release 2.1	This command was implemented on the Cisco ASR 1000 Series Aggregation Services Routers.

Usage Guidelines

After entering the **send** command, the system prompts for the message to be sent, which can be up to 500 characters long. Press **Ctrl-Z** to end the message. Press **Ctrl-C** to terminate this command.

Caution Be aware that in some circumstances text sent using the **send** command may be interpreted as an executable command by the receiving device. For example, if the receiving device is UNIX workstation, and the receiving device is in a state (shell) where commands can be executed, the incoming text, if it is a properly formatted UNIX command, will be accepted by the workstation as a command. For this reason, you should limit your exposure to potential messages from terminal servers or other Cisco IOS-based devices when running an interactive shell.

Examples

The following example shows how to send a message to all lines:

```
Router# send
 *
Enter message, end with CTRL/Z; abort with CTRL/C:
The system 2509 will be shut down in 10 minutes for repairs.^Z
Send message? [confirm]
Router#
***
***
*** Message from tty0 to all terminals:
***
The system 2509 will be shut down in 10 minutes for repairs.
```

Related Commands

Command	Description
reload	Reloads the operating system.

service compress-config

To compress startup configuration files, use the **service compress-config** command in global configuration mode. 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.

Command Default

Disabled

Command Modes

Global configuration

Command History

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

Usage Guidelines

After you configure the **service compress-config** command, the router will compress configuration files every time you save a configuration to the startup configuration. For example, when you enter the **copy system:running-config nvram:startup-config** command, the running configuration will be compressed before storage in NVRAM.

If the file compression succeeds, 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 be compressed enough to fit into NVRAM 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 **compress -b12**command.

Once the configuration file has been compressed, the router functions normally. At boot time, the system recognizes that the configuration file is compressed, uncompresses it, and proceeds normally. A **partition nvram:startup-config** command uncompresses the configuration before displaying it.

To disable compression of the configuration file, enter configuration mode and specify the **no service compress-config** command. Then, exit global configuration mode and enter the **copy system:running-config nvram:startup-config** command. The router displays an OK message if it is able to write the uncompressed configuration to NVRAM. Otherwise, the router displays an error message indicating that the configuration is too large to store. If the configuration file is larger than the physical NVRAM, the following message is displayed:

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

When the file is truncated, commands at the end of the file are erased. Therefore, you will lose part of your configuration. To truncate and save the configuration, type **Y**. To not truncate and not save the configuration, type **N**.

Examples

In the following example, the configuration file is compressed:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# service compress-config
Router(config)# end
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router# copy system:running-config nvram:startup-config
Building configuration...
Compressing configuration from 1179 bytes to 674 bytes
[OK]
```

Related Commands

Command	Description
partition nvram:startup-config	Separates Flash memory into partitions on Class B file system platforms.

service config

To enable autoloading of configuration files from a network server, use the **service config** command in global configuration mode. To restore the default, use the **no** form of this command.

service config
no service config

Syntax Description

This command has no arguments or keywords.

Command Default

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

Command Modes

Global configuration (config)

Command History

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

Usage Guidelines

Usually, the service config command is used in conjunction with the boot host or boot network command. You must enter the service config command to enable the router to automatically configure the system from the file specified by the boot host or boot network command. With Cisco IOS software Releases 12.3(2)T, 12.3(1)B, and later releases, you no longer have to specify the service config command for the boot host or boot network command to be active. If you specify both the no service config command and the boot host command, the router attempts to find the specified host configuration file. The service config command can also be used without the boot host or boot network command. If you do not specify host or network configuration filenames, the router uses the default configuration files. The default network configuration file is network-confg. The default host configuration file is host-confg, where host is the hostname of the router. If the Cisco IOS software cannot resolve its hostname, the default host configuration file is router-confg.

Note You must issue the **reload** command for the **service config** command to take effect.

Examples

In the following example, a router is configured to autoload the default network and host configuration files. Because no **boot host** or **boot network** commands are specified, the router uses the broadcast address to request the files from a TFTP server.

```
Router(config)# service config
```

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

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

Related Commands

Command	Description
boot host	Changes the default name of the host configuration filename from which to load configuration commands.
boot network	Changes the default name of the network configuration file from which to load configuration commands.
Reload	Reloads the operating system.

service counters max age

To set the time interval for retrieving statistics, use the **service counters max age** command in global configuration mode. To return to the default settings, use the **no** form of this command.

service counters max age *seconds*
no service counters max age

Syntax Description

seconds	Specifies the maximum age in seconds to retrieve statistics from the CLI or SNMP. Valid values are from 0 to 60.

Command Default

By default, *seconds* is **0** seconds.

Note For the 6500 and 7600 platforms, a different value is set at system initialization.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(18)SXD	Support for this command was introduced on the Supervisor Engine 720 and the Supervisor Engine 2.
12.2(18)SXF	This command was modified. • The default was changed from 10 seconds to 5 seconds. • The valid values for *seconds* was changed from 1 to 60 seconds to 0 to 60 seconds.
12.2(33)SRA	This command was integrated in Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

A fully loaded Catalyst 6500 series switch chassis that is running Cisco IOS software version 12.2(18)SXF or its minor variants (SXF through SXF5) takes 1 to 2 minutes to update the SNMP counters maintained under ifTable and ifXTable.

To understand the amount of traffic that a specific port/interface handles, the ifTable/ifXTable is polled. The typical polling interval to meet this is 3 to 5 minutes. There is no advantage if you reduce the polling interval to less than 3 minutes.

Note If you decrease the time interval for retrieving statistics from the default setting (5 seconds), traffic congestion results in situations where frequent SNMP (SMNP bulk) retrievals occur.

Examples

This example shows how to set the time interval for retrieving statistics:

```
Router(config)# service counters max age 10
Router(config)#
```

This example shows how to return to the default setting:

```
Router(config)# no service counters max age
Router(config)#
```

service decimal-tty

To specify that line numbers be displayed and interpreted as octal numbers rather than decimal numbers, use the **no service decimal-tty** command in global configuration mode. To restore the default, use the **service decimal-tty** command.

service decimal-tty
no service decimal-tty

Syntax Description This command has no arguments or keywords.

Command Default Enabled (line numbers displayed as decimal numbers)

Command Modes Global configuration

Command History

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

Examples In the following example, the router is configured to display decimal rather than octal line numbers:

```
Router(config)# service decimal-tty
```

service exec-wait

To delay the startup of the EXEC on noisy lines, use the **service exec-wait** command in global configuration mode. To disable the delay function, use the **no** form of this command.

service exec-wait
no service exec-wait

Syntax Description This command has no arguments or keywords.

Command Default Disabled

Command Modes Global configuration

Command History

Release	Modification
10.0	This command was introduced.

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

Usage Guidelines

This command delays startup of the EXEC until the line has been idle (no traffic seen) for 3 seconds. The default is to enable the line immediately on modem activation.

This command is useful on noisy modem lines or when a modem attached to the line is configured to ignore MNP/V.42 negotiations, and MNP/V.42 modems may be dialing in. In these cases, noise or MNP/V.42 packets may be interpreted as usernames and passwords, causing authentication failure before the user has a chance to type a username or password. The command is not useful on nonmodem lines or lines without some kind of login configured.

Examples

The following example delays the startup of the EXEC:

```
Router(config)# service exec-wait
```

service finger

The **service finger** command has been replaced by the **ip finger** command. However, the **service finge**r and **no service finger** commands continue to function to maintain backward compatibility with older versions of Cisco IOS software. Support for this command may be removed in a future release. See the description of the **ip finger** command for more information.

service hide-telnet-address

To hide addresses while trying to establish a Telnet session, use the **service hide-telnet-address** command in global configuration mode. To disable this service, use the **no**form of this command.

service hide-telnet-address
no service hide-telnet-address

Syntax Description

This command has no arguments or keywords.

Command Default

Addresses are displayed.

Command Modes

Global configuration

Command History

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

Usage Guidelines

When you attempt to connect to a device, the router displays addresses and other messages (for example, “Trying router1 (171.69.1.154, 2008)...).” With the hide feature, the router suppresses the display of the address (for example, “Trying router1 address #1...”). The router continues to display all other messages that

would normally be displayed during a connection attempt, such as detailed error messages if the connection was not successful.

The hide feature improves the functionality of the busy-message feature. When you configure only the **busy-message** command, the normal messages generated during a connection attempt are not displayed; only the busy-message is displayed. When you use the hide and busy features together you can customize the information displayed during Telnet connection attempts. When you configure the **service hide-telnet-address** command and the **busy-message** command, the router suppresses the address and displays the message specified with the **busy-message** command if the connection attempt is not successful.

Examples

The following example hides Telnet addresses:

```
Router(config)# service hide-telnet-address
```

Related Commands

Command	Description
busy-message	Creates a “host failed” message that is displayed when a connection fails.

service linenumber

To configure the Cisco IOS software to display line number information after the EXEC or incoming banner, use the **service linenumber** command in global configuration mode. To disable this function, use the **no** form of this command.

service linenumber
no service linenumber

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Global configuration

Command History

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

Usage Guidelines

With the **service linenumber** command, you can have the Cisco IOS software display the host name, line number, and location each time an EXEC process is started, or an incoming connection is made. The line number banner appears immediately after the EXEC banner or incoming banner. This feature is useful for tracking problems with modems, because the host and line for the modem connection are listed. Modem type information can also be included.

Examples

In the following example, a user Telnets to Router2 before and after the **service linenumber** command is enabled. The second time, information about the line is displayed after the banner.

```
Router1> telnet Router2
Trying Router2 (172.30.162.131)... Open
Welcome to Router2.
User Access Verification
Password:
Router2> enable
Password:
Router2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router2(config)# service linenumber
Router2(config)# end
Router2# logout
[Connection to Router2 closed by foreign host]
Router1> telnet Router2
Trying Router2 (172.30.162.131)... Open
Welcome to Router2.
Router2 line 10
User Access Verification
Password:
Router2>
```

Related Commands

Command	Description
show users	Displays information about the active lines on the router.

service nagle

To enable the Nagle congestion control algorithm, use the **service nagle** command in global configuration mode. To disable the algorithm, use the **no** form of this command.

service nagle
no service nagle

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Global configuration

Command History

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

Usage Guidelines

When using a standard TCP implementation to send keystrokes between machines, TCP tends to send one packet for each keystroke typed. On larger networks, many small packets use up bandwidth and contribute to congestion.

The algorithm developed by John Nagle (RFC 896) helps alleviate the small-packet problem in TCP. In general, it works this way: The first character typed after connection establishment is sent in a single packet, but TCP holds any additional characters typed until the receiver acknowledges the previous packet. Then the second, larger packet is sent, and additional typed characters are saved until the acknowledgment comes back.

The effect is to accumulate characters into larger chunks, and pace them out to the network at a rate matching the round-trip time of the given connection. This method is usually effective for all TCP-based traffic. However, do not use the **service nagle** command if you have XRemote users on X Window system sessions.

Examples

The following example enables the Nagle algorithm:

```
Router(config)# service nagle
```

service prompt config

To display the configuration prompt (config), use the **service prompt config** command in global configuration mode. To remove the configuration prompt, use the **no** form of this command.

service prompt config
no service prompt config

Syntax Description

This command has no arguments or keywords.

Command Default

The configuration prompts appear in all configuration modes.

Command Modes

Global configuration

Command History

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

Examples

In the following example, the **no service prompt config** command prevents the configuration prompt from being displayed. The prompt is still displayed in EXEC mode. When the **service prompt config** command is entered, the configuration mode prompt reappears.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# no service prompt config
hostname newname
end
newname# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
service prompt config
newname(config)# hostname Router
Router(config)# end
Router#
```

Related Commands

Command	Description
hostname	Specifies or modifies the host name for the network server.
prompt	Customizes the prompt.

service sequence-numbers

To enable visible sequence numbering of system logging messages, use the **service sequence-numbers** command in global configuration mode. To disable visible sequence numbering of logging messages, use the **no** form of this command.

service sequence-numbers
no service sequence-numbers

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Global configuration

Command History

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

Usage Guidelines

Each system status messages logged in the system logging process have a sequence reference number applied. This command makes that number visible by displaying it with the message. The sequence number is displayed as the first part of the system status message. See the description of the **logging** commands for information on displaying logging messages.

Examples

In the following example logging message sequence numbers are enabled:

```
.Mar 22 15:28:02 PST: %SYS-5-CONFIG_I: Configured from console by console
Router# config terminal

Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# service sequence-numbers

Router(config)# end

Router#
000066: .Mar 22 15:35:57 PST: %SYS-5-CONFIG_I: Configured from console by console
```

Related Commands

Command	Description
logging on	Enables system logging globally.
service timestamps	Enables time-stamping of system logging messages or debugging messages.

service slave-log

To allow secondary Versatile Interface Processor (VIP) cards to log important error messages to the console, use the **service slave-log** command in global configuration mode . To disable secondary logging, use the **no** form of this command.

service slave-log
no service slave-log

Syntax Description This command has no arguments or keywords.

Command Default This command is enabled by default.

Command Modes Global configuration

Command History

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

Usage Guidelines This command allows secondary slots to log error messages of level 2 or higher (critical, alerts, and emergencies).

Examples In the following example, the router is configured to log important messages from the secondary cards to the console:

```
Router(config)# service slave-log
```

The following is sample output generated when this command is enabled:

```
%IPC-5-SLAVELOG: VIP-SLOT2:
 IPC-2-NOMEM: No memory available for IPC system initialization
```

The first line indicates which slot sent the message. The second line contains the error message.

service tcp-keepalives-in

To generate keepalive packets on idle incoming network connections (initiated by the remote host), use the **service tcp-keepalives-in** command in global configuration mode . To disable the keepalives, use the **no**form of this command.

service tcp-keepalives-in
no service tcp-keepalives-in

Syntax Description This command has no arguments or keywords.

Command Default Disabled

Command Modes Global configuration

Command History

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

Examples

In the following example, keepalives on incoming TCP connections are generated:

```
Router(config)# service tcp-keepalives-in
```

Related Commands

Command	Description
service tcp-keepalives-out	Generates keepalive packets on idle outgoing network connections (initiated by a user).

service tcp-keepalives-out

To generate keepalive packets on idle outgoing network connections (initiated by a user), use the **service tcp-keepalives-out** command in global configuration mode . To disable the keepalives, use the **no**form of this command.

service tcp-keepalives-out
no service tcp-keepalives-out

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled

Command Modes

Global configuration

Command History

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

Examples

In the following example, keepalives on outgoing TCP connections are generated:

```
Router(config)# service tcp-keepalives-out
```

Related Commands

Command	Description
service tcp-keepalives-in	Generates keepalive packets on idle incoming network connections (initiated by the remote host).

service tcp-small-servers

To enable small TCP servers such as the Echo, use the **service tcp-small-servers** command in global configuration mode. To disable the TCP server, use the **no** form of this command.

service tcp-small-servers [{**max-servers** *number* | **no-limit**}]
no service tcp-small-servers [{**max-servers** *number* | **no-limit**}]

Syntax Description

max-servers	(Optional) Sets the number of allowable TCP small servers.
number	(Optional) Maximum number of TCP small servers. Range is 1 to 2147483647.
no-limit	(Optional) Allows the number of TCP small servers to have no limit.

Command Default

TCP small servers are disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.
12.2(33)SRC	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SRC.
12.2(33)SXI	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SXI.
Cisco IOS XE Release 2.1	This command was implemented on the Cisco ASR 1000 Series Aggregation Services Routers.

Usage Guidelines

To use the **service tcp-small-servers** command, you must be in a user group associated with a task group that includes the proper task IDs. If you suspect user group assignment is preventing you from using a command, contact your Authentication, Authorization, and Accounting (AAA) administrator for assistance.

The TCP small servers consist of three services: Discard (port 9), Echo (port 7), and Chargen (port 19). These services are used to test the TCP transport functionality. The discard server receives data and discards it. The echo server receives data and echoes the same data to the sending host. The chargen server generates a sequence of data and sends it to the remote host.

Examples

The following example shows how to enable small TCP servers and set the maximum number of allowable small servers to 14:

```
Router(config)#
 service tcp-small-servers max-servers 14
```

Related Commands

Command	Description
service udp-small-servers	Enables small UDP servers such as the Echo.

service telnet-zeroidle

To set the TCP window to zero (0) when the Telnet connection is idle, use the **service telnet-zeroidle** command in global configuration mode. To disable this service, use the **no** form of this command.

service telnet-zero-idle
no service telnet-zeroidle

Syntax Description

This command has no arguments or keywords.

Command Default

The TCP window is not set to zero when the the Telnet connection is idle.

Command Modes

Global configuration (config)

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.
12.2(33)SRC	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SRC.

Usage Guidelines

Normally, data sent to noncurrent Telnet connections is accepted and discarded. When the **service telnet-zero-idle** command is enabled, if a session is suspended (that is, some other connection is made active or the router is in the privileged EXEC mode), the TCP window is set to zero. This action prevents the remote host from sending any more data until the connection is resumed. Use this command when it is important that all messages sent by the host be seen by the users and the users are likely to use multiple sessions.

Do not use this command if your host will eventually time out and log out a TCP user whose window is zero.

Examples

The following example shows how to set the TCP window to zero when the Telnet connection is idle:

```
Router(config)# service telnet-zeroidle
```

Related Commands

Command	Description
resume	Switches to another open Telnet, rlogin, LAT, or PAD session.

service timestamps

To configure the system to apply a time stamp to debugging messages or system logging messages, use the **service timestamps** command in global configuration mode . To disable this service, use the **no** form of this command.

service timestamps [{**debug** | **log**}] [{**uptime** | **datetime** [**msec**]}] [**localtime**] [**show-timezone**] [**year**]
no service timestamps [{**debug** | **log**}]

Syntax Description

debug	(Optional) Indicates time-stamping for debugging messages.
log	(Optional) Indicates time-stamping for system logging messages.
uptime	(Optional) Specifies that the time stamp should consist of the time since the system was last rebooted. For example "4w6d" (time since last reboot is 4 weeks and 6 days). • This is the default time-stamp format for both debugging messages and logging messages. • The format for uptime varies depending on how much time has elapsed: • *HHHH* :*MM* :*SS* (*HHHH* hours: *MM* minutes: *SS* seconds) for the first 24 hours • *D* d*HH* h (*D* days *HH* hours) after the first day • *W* w*D* d (*W* weeks *D* days) after the first week
datetime	(Optional) Specifies that the time stamp should consist of the date and time. • The time-stamp format for **datetime** is MMM DD HH:MM:SS, where MMM is the month, DD is the date, HH is the hour (in 24-hour notation), MM is the minute, and SS is the second. • If the datetime keyword is specified, you can optionally add the **msec localtime**, **show-timezone**, or **year** keywords. • If the **service timestamps datetime** command is used without addtional keywords, time stamps will be shown using UTC, without the year, without milliseconds, and without a time zone name.
msec	(Optional) Includes milliseconds in the time stamp, in the format *HH***:***DD***:***MM***:***SS***.***mmm* , where *.mmm* is milliseconds
localtime	(Optional) Time stamp relative to the local time zone.
year	(Optional) Include the year in the date-time format.
show-timezone	(Optional) Include the time zone name in the time stamp. **Note** If the localtime keyword option is not used (or if the local time zone has not been configured using the **clock timezone** command), time will be displayed in Coordinated Universal Time (UTC).

Command Default

Time stamps are applied to debug and logging messages.

Command Modes

Global configuration (config)

Command History

Release	Modification
10.0	This command was introduced.
11.3(5)	Service time stamps are enabled by default.
12.3(1)	The **year** keyword was added.

Release	Modification
12.3(2)T	This command was integrated into Cisco IOS Release 12.3(2)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Usage Guidelines

Time stamps can be added to either debugging messages (**service timestamp debug**) or logging messages (**service timestamp log**) independently.

If the **service timestamps**command is specified with no arguments or keywords, the default is **service timestamps debug uptime**.

The **no service timestamps**command by itself disables time stamps for both debug and log messages.

The **uptime** form of the command adds time stamps (such as "2w3d") that indicating the time since the system was rebooted. The **datetime** form of the command adds time stamps (such as "Sep 5 2002 07:28:20") that indicate the date and time according to the system clock.

Entering the **service timestamps** {**debug** | **log**} command a second time will overwrite any previously configured **service timestamp** {**debug** | **log**} commands and associated options.

To set the local time zone, use the **clock timezone***zonehours-offset* command in global configuration mode.

The time stamp will be preceeded by an asterisk or period if the time is potentially inaccurate. The table below describes the symbols that proceed the time stamp.

Table 1: Time-Stamping Symbols for syslog Messages

Symbol	Description	Example
(blank)	Time is authoritative: the software clock is in sync or has just been set manually	15:29:03.158 UTC Tue Feb 25 2003:
*	Time is not authoritative: the software clock has not been set, or is not in sync with configured Network Time Protocol (NTP) servers.	*15:29:03.158 UTC Tue Feb 25 2003:
.	Time is authoritative, but the NTP is not synchronized: the software clock was in sync, but has since lost contact with all configured NTP servers.	.15:29:03.158 UTC Tue Feb 25 2003:

Examples

In the following example, the router begins with time-stamping disabled. Then, the default time-stamping is enabled (uptime time stamps applied to debug output). Then, the default time-stamping for logging is enabled (uptime time stamps applied to logging output).

```
Router# show running-config | include time

no service timestamps debug uptime
no service timestamps log uptime
Router# config terminal
```

```
Router(config)# service timestamps

! issue the show running-config command in config mode using do Router(config)# do show
running-config | inc time

! shows that debug timestamping is enabled, log timestamping is disabled
service timestamps debug uptime
no service timestamps log uptime
! enable timestamps for logging messages
Router(config)# service timestamps log
Router(config)# do show run | inc time

service timestamps debug uptime
service timestamps log uptime
Router(config)# service sequence-numbers

Router(config)# end

000075: 5w0d: %SYS-5-CONFIG_I: Configured from console by console
! The following is a level 5 system logging message
! The leading number comes from the service sequence-numbers command.
! 4w6d indicates the timestamp of 4 weeks, 6 days000075: 4w6d: %SYS-5-CONFIG_I: Configured
 from console by console
```

In the following example, the user enables time-stamping on logging messages using the current time and date in Coordinated Universal Time/Greenwich Mean Time (UTC/GMT), and enables the year to be shown.

```
Router(config)#
! The following line shows the timestamp with uptime (1 week 0 days)
1w0d: %SYS-5-CONFIG_I: Configured from console by console
Router(config)# service timestamps log datetime show-timezone
 year

Router(config)# end
! The following line shows the timestamp with datetime (11:13 PM March 22nd)
.Mar 22 2004 23:13:25 UTC: %SYS-5-CONFIG_I: Configured from console by console
```

The following example shows the change from UTC to local time:

```
Router# configure terminal

! Logging output can be quite long; first changing line width to show full
! logging message
Router(config)# line 0

Router(config-line)# width 180

Router(config-line)# logging synchronous

Router(config-line)# end

! Timestamping already enabled for logging messages; time shown in UTC.
Oct 13 23:20:05 UTC: %SYS-5-CONFIG_I: Configured from console by console
Router# show clock

23:20:53.919 UTC Wed Oct 13 2004
Router# configure terminal

Enter configuration commands, one per line.  End with the end command.
! Timezone set as Pacific Standard Time, with an 8 hour offset from UTC
Router(config)# clock timezone PST -8
```

```
Router(config)#
Oct 13 23:21:27 UTC: %SYS-6-CLOCKUPDATE:
System clock has been updated from 23:21:27 UTC Wed Oct 13 2004
to 15:21:27 PST Wed Oct 13 2004, configured from console by console.
Router(config)#
! Pacific Daylight Time (PDT) configured to start in April and end in October.
! Default offset is +1 hour.
Router(config)# clock summer-time PDT recurring first Sunday April 2:00 last Sunday October
 2:00
Router(config)#
! Time changed from 3:22 P.M. Pacific Standard Time (15:22 PST)

! to 4:22 P.M. Pacific Daylight (16:22 PDT)

Oct 13 23:22:09 UTC: %SYS-6-CLOCKUPDATE:
System clock has been updated from 15:22:09 PST Wed Oct 13 2004
to 16:22:09 PDT Wed Oct 13 2004, configured from console by console.
! Change the timestamp to show the local time and timezone.
Router(config)# service timestamps log datetime localtime show-timezone

Router(config)# end

Oct 13 16:23:19 PDT: %SYS-5-CONFIG_I: Configured from console by console
Router# show clock

16:23:58.747 PDT Wed Oct 13 2004
Router# config t

Enter configuration commands, one per line.  End with the end command.
Router(config)# service sequence-numbers

Router(config)# end

Router#
```

In the following example, the **service timestamps log datetime** command is used to change previously configured options for the date-time time stamp.

```
Router(config)# service timestamps log datetime localtime show-timezone
Router(config)# end
! The year is not displayed.
Oct 13 15:44:46 PDT: %SYS-5-CONFIG_I: Configured from console by console
Router# config t

Enter configuration commands, one per line.  End with the end command.
Router(config)# service timestamps log datetime show-timezone year

Router(config)# end

! note: because the
localtime option was not specified again, that option is

! removed from the output, and time is displayed in UTC (the default)

Oct 13 2004 22:45:31 UTC: %SYS-5-CONFIG_I: Configured from console by console
```

Related Commands

Command	Description
clock set	Manually sets the system clock.

Command	Description
ntp	Controls access to the system's NTP services.
service sequence-numbers	Stamps system logging messages with a sequence number.

service udp-small-servers

To enable small User Datagram Protocol (UDP) servers such as the Echo, use the **service udp-small-servers** command in global configuration mode. To disable the UDP server, use the **no** form of this command.

service udp-small-servers [{**max-servers** *number* | **no-limit**}]
no service udp-small-servers [{**max-servers** *number* | **no-limit**}]

Syntax Description

max-servers	(Optional) Sets the number of allowable UDP small servers.
number	(Optional) Maximum number of UDP small servers. Range is 1 to 2147483647.
no-limit	(Optional) Allows the number of TCP small servers to have no limit.

Command Default

UDP small servers are disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(1)M	This command was introduced in a release earlier than Cisco IOS Release 15.0(1)M.
12.2(33)SRC	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SRC.
12.2(33)SXI	This command was integrated into a release earlier than Cisco IOS Release 12.2(33)SXI.
Cisco IOS XE Release 2.1	This command was implemented on the Cisco ASR 1000 Series Aggregation Services Routers.

Usage Guidelines

To use this command, you must be in a user group associated with a task group that includes the proper task IDs. If you suspect user group assignment is preventing you from using a command, contact your Authentication, Authorization, and Accounting (AAA) administrator for assistance.

The UDP small servers currently consist of three services: Discard (port 9), Echo (port 7), and Chargen (port 19). These services are used to test the UDP transport functionality. The discard server receives data and discards it. The echo server receives data and echoes the same data to the sending host. The chargen server generates a sequence of data and sends it to the remote host.

Examples

The following example shows how to enable small UDP servers and set the maximum number of allowable small servers to 10:

```
Router(config)#
 service udp-small-servers max-servers 10
```

Related Commands

Command	Description
service tcp-small-servers	Enables small TCP servers such as the Echo.

service-module apa traffic-management

To configure traffic management on the router, use the **service-module apa traffic-management**command in interface configuration mode.

service-module apa traffic-management [{**monitor** | **inline**}]

Syntax Description

monitor	Enables promiscuous monitoring.
inline	Enables inline monitoring.

Command Default

None

Command Modes

Interface configuration mode

Command History

Release	Modification
12.4(20)YA	This command was introduced for the NME-APA on Cisco 2811, 2821, 2851, and Cisco 3800 Series Integrated Services Routers.

Usage Guidelines

To perform traffic management, you enable or disable the flow of packets by configuring the service module interface and the router interface.

- Configure the router interface with the **service-module apa traffic-management [monitor | inline]**command.

Two traffic management options are available:

- - Monitor--will copy the packet and designate the copy as the one forwarded to the Application Performance Assurance module (NME-APA).
  - Inline--will send the packet to the NME-APA, rather than sending a copy of the packet. After the NME-APA has processes the packet, it sends it back to the router.

Note Enable only one traffic management option on the router, but not both concurrently.

- Configure the service module interface with the Application Performance Assurance (APA) graphical user interface (GUI). See the *Cisco Application Performance Assurance User Guide* for details.

Examples

The following example configures an interface on a Cisco 2851 Integrated Services Router for inline traffic management.

```
Router> enable
Router# configure terminal
Router(config)# interface gigabitethernet 0/1
Router(config-if)# ip address
 10.10.10.43 255.255.255.0
Router(config-if)# service-module apa traffic-management inline
Router(config-if)# exit
```

end

Related Commands

Command	Description
interface gigabitethernet	Defines the interface on the router
ip address	Defines the IP address and subnet mask on the interface

service-module wlan-ap bootimage

To configure the boot image on the service module, use the **service-module wlan-ap bootimage** command in privileged EXEC mode.

service-module wlan-ap *interface number* **bootimage** [{**autonomous** | **unified**}]

Syntax Description

interface number	The interface number for the wireless device. Always use 0.
autonomous	Autonomous software image.
unified	Upgrade image with Lightweight Access Point Protocol (LWAPP).

Command Default

Autonomous software image

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(20) T	This command was introduced for wireless-enabled Cisco 880 Series and Cisco 890 Series Integrated Services Routers.

Usage Guidelines

When running the advanced IP services feature set on either Cisco 880 Series routers or Cisco 890 Series routers, use the **service-module wlan-ap 0 bootimage unified command**to enable the Cisco unified software upgrade image on the embedded wireless access point. After enabling the unified image, use the **service-module wlan-ap 0 reload** command to perform a graceful shutdown and reboot of the access point.

Note The **service-module wlan-ap 0 bootimage**command does not support recovery images on the embedded access point. Use the **service-module wlan-ap 0 reload** command to shutdown and reboot the access point.

Cisco 880 Series and Cisco 890 Series routers with embedded access point running the unified software image require DHCP to obtain an IP address for the access point. An IP address is needed to communicate with the Wireless LAN Controller (WLC) and to download its image upon boot up. The host router can provide DHCP server functionality through the DHCP pool to reach the WLC, and setup option 43 for the controller IP address in the DHCP pool configuration.

Use the following guideline to setup a DHCP pool on the host router.

```
ip dhcp pool embedded-ap-pool
    network 60.0.0.0 255.255.255.0
    default router 60.0.0.1
    option 43 hex f104.0a0a.0a0f  /* Single WLC IP address (10.10.10.15) in HEX format */
int vlan 1  /* Default Vlan */
ip address 60.0.0.1 255.255.255.0
int Wlan-GigabitEthernet0 /* internal switch-port to AP */
switchport access vlan 1
```

Examples

The following example upgrades the embedded access point image from autonomous to unified.

```
Router#configure terminal
Router(config)#service-module wlan-ap 0 bootimage unified
 *Jan 18 05:31:58.172: %WLAN_AP_SM-6-UNIFIED_IMAGE: Embedded AP will change boot image to
mini-IOS also called LWAPP recovery Please check router config to ensure connectivity between
 WLC and AP. Use service-module wlan-ap 0 reload to bootup mini-IOS image on AP
Router(config)#end
Router#
*Jan 18 05:32:04.136: %SYS-5-CONFIG_I: Configured from console by console
Router#service-module wlan-ap 0 reload
 Reload will save AP config....
Do you want to proceed with reload?[confirm] Trying to reload Service Module wlan-ap0.
Router#
Service Module saved config, start reset.
Received reload request from router
Saving configuration...
Building configuration...
```

Related Commands

Command	Description
interface wlan-ap	Enters wireless interface configuration mode to configure an interface.
service-module wlan-ap reload	Performs a graceful shutdown and reboot of the service module.
service-module wlan-ap reset	Resets the service module hardware.

service-module wlan-ap reload

To perform a graceful shutdown and reboot of the service module use the **service-module wlan-ap reload** command in privileged EXEC mode.

service-module wlan-ap *interface number* **reload**

Syntax Description

interface number	The interface number for the wireless device. Always use 0.

Command Default

None

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(20)T	This command was introduced for wireless-enabled Cisco 860, 880, and 890 Integrated Services Routers.

Usage Guidelines

Autonomous Mode

At the confirmation prompt, press **Enter** to confirm the action, or press **n** to cancel.

Note When running in autonomous mode, the reload command saves the configuration before rebooting. If the attempt is unsuccessful, the following message displays: Failed to save service module configuration.

Unified Mode

The service module reload command is usually handled by the Wireless LAN Controller (WLC).

Note When running in Unified mode, the reload command will produce the following message: The embedded wireless device is in Unified mode. Reload/reset is normally handled by WLC controller. Still want to proceed? [yes]

Examples

The following examples show a graceful shut down and reboot of the service module:

Autonomous Mode

```
Router# service-module wlan-ap0 reload
Do you want to proceed with reload?[confirm]
Router# reload
Do you want to reload the internal AP ? [yes/no]:
Do you want to save the configuration of the AP ? [yes/no]:
System configuration has been modified. Save [yes/no]:
Proceed with reload? [confirm]
```

Unified Mode

```
Router# service-module wlan-ap0 reload
The embedded AP is in Unified mode. Reload/reset is normally handled by WLC controller.
Still want to proceed? [yes]
Router# reload
The embedded AP is in Unified mode. Reload/reset is normally handled by WLC controller.
Do you want to reload the internal AP [yes/no]:
System configuration has been modified. Save [yes/no]:
Proceed with reload [Confirm]
```

Related Commands

Command	Description
interface wlan-ap	Enters wireless interface configuration mode to configure an interface.
service-module wlan-ap reset	Resets the service module hardware.

service-module wlan-ap reset

To reset the service module hardware, software, and configuration, use the **service-module wlan-ap reset** command in privileged EXEC mode.

service-module wlan-ap interface number reset [{**bootloader** | **default-config**}]

Syntax Description

interface number	The interface number for the wireless device. Always use 0.
bootloader	Resets the wireless device to the bootloader for manual image recovery.
default-config	Resets the wireless device to the factory default configuration.

Command Default

None

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(20)T	This command was introduced for wireless-enabled Cisco 860, 880, and 890 Integrated Services Routers.

Usage Guidelines

At the confirmation prompt, press **Enter** to confirm the action, or press **n** to cancel.

Caution Because you may lose data, use the **service-module wlan-ap reset** command only to recover from a shutdown or failed state.

Examples

The following example resets a wireless device on a router that is operating in either autonomous mode or LWAPP mode:

Autonomous Mode

```
Router# service-module wlan-ap0 reset
Use reset only to recover from shutdown or failed state.
```

LWAPP Mode

```
Router# service-module wlan-ap0 reset
```

```
The embedded device is in LWAPP mode. Reload/reset is normally handled by WLC controller.
Still want to proceed? [yes]
```

Resetting the Factory Default Configuration on the Wireless Device

The following example resets the wireless device to the default configuration.

```
Router#service-module wlan-ap 0 reset default-config
Router#
```

Recovering the Image on the Wireless Device

The following example resets the wireless device down to the bootloader level for manual image recovery.

```
Router#service-module wlan-ap0 reset bootloader
Router#
```

Related Commands

Command	Description
interface wlan-ap	Enters wireless interface configuration mode to configure an interface.
service-module wlan-ap reload	Performs a graceful shutdown and reboot of the service module.

service-module wlan-ap session

To begin a configuration session with a service module through a console connection use the **service-module wlan-ap session** command in privileged EXEC mode.

service-module wlan-ap *interface number* **session** [{**clear** | **disconnect**}]

Syntax Description

interface number	The interface number for the wireless device. Always use 0.
clear	(Optional) Clears the wireless device configuration session.

Command Default

None

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(20)T	This command was introduced for wireless-enabled Cisco 860, 880, and 890 Integrated Services Routers.

Usage Guidelines

Only one session is allowed at a time into the wireless device from a router console-port connection. After starting a session, perform configuration tasks on the wireless device. You first access the router in a user-level

shell. To access the privileged EXEC command shell, where most commands are available, use the **enable** command.

When you finish configuring the device, and would like to exit the console session, type Ctrl-Shift 6x to return to the router's console. Type service-module wlan-ap session **clear** or disconnect to close the session with the device. At the confirmation prompt, press **Enter twice**to confirm the action or **n** to cancel.

Note If you do not clear or disconnect the session on the service module, it will remain open in the background after you return to the router's console prompt. When the session is open in the background, pressing Enter will toggle you back to the wireless device prompt.

Examples

The following example shows a session being opened on a service-module in an ISR:

```
Router# service-module wlan-ap 0 session
Trying 1.2.3.4, 2002 ... Open
AP#
```

The following example clears the session on the service-module in the ISR:

```
Router#service-module wlan-ap 0 session clear
[confirm]
 [OK]
```

Related Commands

Command	Description
enable	Enters privileged EXEC mode.
interface wlan-ap	Enters wireless interface configuration mode to configure an interface.

service-module wlan-ap statistics

To display reset and reload information for a service module and its operating system software, use the **service-module wlan-ap statistics**command in privileged EXEC mode.

service-module wlan-ap interface number statistics

Syntax Description

interface number	The interface number for the wireless device. Always use 0.

Command Default none

Command Modes Privileged EXEC

Command History

Release	Modification
12.4(20)T	This command was introduced for wireless-enabled Cisco 860, 880, and 890 Integrated Services Routers.

Examples

The following example displays information for wireless-enabled Cisco ISRs:

```
Router#service-module wlan-ap 0 statistics
Module Reset Statistics:
  CLI reset count = 0
  CLI reload count = 1
  Registration request timeout reset count = 0
  Error recovery timeout reset count = 0
  Module registration count = 10
The last IOS initiated event was a cli reload at *04:27:32.041 UTC Fri Mar 8 2007
```

Related Commands

Command	Description
interface wlan-ap	Enters wireless interface configuration mode and configures a wireless device.
service-module wlan-ap reset	Resets the wireless device.
service-module wlan-ap reload	Performs a graceful shutdown and reboot on the wireless device.

service-module wlan-ap status

To display configuration information related to hardware and software on the service module, use the **service-module wlan-ap status**command in privileged EXEC mode.

service-module wlan-ap *interface number* **status**

Syntax Description

interface number	The interface number for the wireless device. Aways use 0.

Command Default

None

Command Modes

Privileged EXEC

Command History

Release	Modification
12.4(20)T	This command was introduced for wireless-enabled Cisco 860, 880, and 890 Integrated Services Routers.

Usage Guidelines

Use the **service-module wlan-ap status** command to

- Display the wireless device's software release version
- Check the wireless device's status (steady or down)
- Display hardware information for the wireless device, including image, memory, interface, and system uptime

Examples

The following example displays information for the wireless device on a Cisco Integrated Services Router:

```
Router#service-module wlan-ap 0 status

Service Module is Cisco wlan-ap0
Service Module supports session via TTY line 2
Service Module is in Steady state
Service Module reset on error is disabled
Getting status from the Service Module, please wait..
Image path = flash:c8xx_19xx_ap-k9w7-mx.acregr/c8xx_19xx_ap-k9w7-mx.acre
gr
System uptime = 0 days, 4 hours, 28 minutes, 5 seconds
Router#d was introduced for embedded wireless LAN access points on Cisco 860 and 880 Series
 Integrated Services Routers.
```

Related Commands

Command	Description
interface wlan-ap	Enters wireless service module's console interface.

session slot

To open a session with a module (for example, the Multilayer Switch Module (MSM), Network Analysis Module (NAM), or Asynchronous Transfer Mode (ATM)), use the **session slot** command in EXEC mode.

session slot *mod* **processor** *processor-id*

Syntax Description

mod	Slot number.
processor *processor-id*	Specifies the processor ID.

Command Default

This command has no default settings.

Command Modes

EXEC

Command History

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

Usage Guidelines

To end the session, enter the **quit** command.

This command allows you to use the module-specific CLI.

Examples

This example shows how to open a session with an MSM (module 4):

```
Router# session slot 4 processor 2
Router#
```

set memory debug incremental starting-time

To set the current time as the starting time for incremental analysis, use the **set memory debug incremental starting-time** command in privileged EXEC mode.

set memory debug incremental starting-time [**none**]

Syntax Description

none	(Optional) Resets the defined start time for incremental analysis.

Command Default

No default behavior or values.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(8)T1	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

For incremental analysis, a starting point can be defined by using the **set memory debug incremental starting-time** command. When a starting time is set, only memory allocated after that starting time will be considered for reporting as leaks.

Examples

The following example shows the command used to set the starting time for incremental analysis to the time when the command was issued:

```
Router# set memory debug incremental starting-time
```

Related Commands

Command	Description
show memory debug incremental allocation	Displays all memory blocks that were allocated after the issue of the **set memory debug incremental starting-time** command.
show memory debug incremental leaks	Displays only memory that was leaked after the issue of the **set memory debug incremental starting-time** command.
show memory debug incremental leaks lowmem	Forces incremental memory leak detection to work in low memory mode. Displays only memory that was leaked after the issue of the **set memory debug incremental starting-time** command.
show memory debug incremental status	Displays if the starting point of incremental analysis has been defined and the time elapsed since then.
show memory debug leaks	Displays detected memory leaks.

setup

To enter Setup mode, use the **setup** command in privileged EXEC mode.

setup

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

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

Usage Guidelines

Setup mode gives you the option of configuring your system without using the Cisco IOS Command Line Interface (CLI). For some tasks, you may find it easier to use Setup than to enter Cisco IOS commands individually. For example, you might want to use Setup to add a protocol suite, to make major addressing scheme changes, or to configure a newly installed interface. Although you can use the CLI to make these changes, Setup provides you with a high-level view of the configuration and guides you through the configuration process.

If you are not familiar with Cisco products and the CLI, Setup is a particularly valuable tool because it prompts you for the specific information required to configure your system.

Note If you use the Setup mode to modify a configuration because you have added or modified the hardware, be sure to verify the physical connections using the **show version** EXEC command. Also, verify the logical port assignments using the **show running-config** EXEC command to ensure that you configure the correct port. Refer to the hardware documentation for your platform for more information on physical and logical port assignments.

Before using the Setup mode, you should have the following information so that you can configure the system properly:

- Which interfaces you want to configure
- Which routing protocols you wish to enable
- Whether the router is to perform bridging
- Network addresses for the protocols being configured
- Password strategy for your environment

When you enter the **setup** EXEC command after first-time startup, an interactive dialog called the *System Configuration Dialog* appears on the system console screen. The System Configuration Dialog guides you through the configuration process. It prompts you first for global parameters and then for interface parameters. The values shown in brackets next to each prompt reflect either the default settings or the last configured setting.

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

You must progress through the System Configuration Dialog until you come to the item that you intend to change. To accept default settings for items that you do not want to change, press the **Return** or **Enter** key. The default choice is indicated by square brackets (for example, [yes]) before the prompt colon (:).

To exit Setup mode and return to privileged EXEC mode without making changes and without progressing through the entire System Configuration Dialog, press **Ctrl-C**

The facility also provides help text for each prompt. To access help text, press the question mark (**?**) key at a prompt.

When you complete your changes, the system will automatically display the configuration file that was created during the Setup session. It also asks you if you want to use this configuration. If you answerYes, the configuration is saved to NVRAM as the startup configuration file. If you answer No, the configuration is not saved and the process begins again. There is no default for this prompt; you must answer either Yes or No.

Examples

The following example displays the **setup** command facility to configure serial interface 0 and to add ARAP and IP/IPX PPP support on the asynchronous interfaces:

```
Router# setup
         --- System Configuration Dialog
 ---
At any point you may enter a question mark '?' for help.
Use ctrl-c to
abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Continue with configuration dialog? [yes]:
First, would you like to see the current
interface summary? [yes]:
Interface          IP-Address      OK?  Method    Status                  Protocol
Ethernet0          172.16.72.2     YES  manual    up                      up
Serial0            unassigned      YES  not set   administratively down  down
Serial1            172.16.72.2     YES  not set   up                      up
Configuring global parameters:
  Enter host name [Router]:
The enable secret is a one-way cryptographic secret used
instead of the enable password when it exists.
  Enter enable secret [<Use current secret>]:


The enable password is used when there is no enable secret
and when using older software and some boot images.


  Enter enable password [ww]:
  Enter virtual terminal password [ww]:
  Configure SNMP Network Management? [yes]:
    Community string [public]:
  Configure DECnet? [no]:
  Configure AppleTalk? [yes]:
    Multizone networks? [no]: yes
  Configure IPX? [yes]:
  Configure IP? [yes]:
    Configure IGRP routing? [yes]:
      Your IGRP autonomous system number [15]:
  Configure Async lines? [yes]:
    Async line speed [9600]: 57600
    Configure for HW flow control? [yes]:
    Configure for modems? [yes/no]: yes
```

```
      Configure for default chat script? [yes]: no
    Configure for Dial-in IP SLIP/PPP access? [no]: yes
      Configure for Dynamic IP addresses? [yes]: no

      Configure Default IP addresses? [no]: yes
      Configure for TCP Header Compression? [yes]: no
      Configure for routing updates on async links? [no]:
    Configure for Async IPX? [yes]:
    Configure for Appletalk Remote Access? [yes]:
      AppleTalk Network for ARAP clients [1]: 20
      Zone name for ARAP clients [ARA Dialins]:
Configuring interface parameters:
Configuring interface Ethernet0:
  Is this interface in use? [yes]:
  Configure IP on this interface? [yes]:
    IP address for this interface [172.16.72.2]:
    Number of bits in subnet field [8]:
    Class B network is 172.16.0.0, 8 subnet bits; mask is /24
  Configure AppleTalk on this interface? [yes]:
    Extended AppleTalk network? [yes]:
    AppleTalk starting cable range [1]:
    AppleTalk ending cable range [1]:
    AppleTalk zone name [Sales]:
    AppleTalk additional zone name:
  Configure IPX on this interface? [yes]:
    IPX network number [1]:
Configuring interface Serial0:
  Is this interface in use? [no]: yes
  Configure IP on this interface? [no]: yes
  Configure IP unnumbered on this interface? [no]: yes
    Assign to which interface [Ethernet0]:
  Configure AppleTalk on this interface? [no]: yes
    Extended AppleTalk network? [yes]:
    AppleTalk starting cable range [2]: 3
    AppleTalk ending cable range [3]: 3
    AppleTalk zone name [myzone]: ZZ Serial
    AppleTalk additional zone name:
  Configure IPX on this interface? [no]: yes
    IPX network number [2]: 3
Configuring interface Serial1:
  Is this interface in use? [yes]:
  Configure IP on this interface? [yes]:
  Configure IP unnumbered on this interface? [yes]:
    Assign to which interface [Ethernet0]:
  Configure AppleTalk on this interface? [yes]:
    Extended AppleTalk network? [yes]:
    AppleTalk starting cable range [2]:
    AppleTalk ending cable range [2]:
    AppleTalk zone name [ZZ Serial]:
    AppleTalk additional zone name:
  Configure IPX on this interface? [yes]:
    IPX network number [2]:
Configuring interface Async1:
    IPX network number [4]:
    Default client IP address for this interface [none]: 172.16.72.4
Configuring interface Async2:
    IPX network number [5]:
    Default client IP address for this interface [172.16.72.5]:
Configuring interface Async3:
    IPX network number [6]:
    Default client IP address for this interface [172.16.72.6]:
Configuring interface Async4:
    IPX network number [7]:
    Default client IP address for this interface [172.16.72.7]:
```

```
Configuring interface Async5:
    IPX network number [8]:
    Default client IP address for this interface [172.16.72.8]:
Configuring interface Async6:
    IPX network number [9]:
    Default client IP address for this interface [172.16.72.9]:
Configuring interface Async7:
    IPX network number [A]:
    Default client IP address for this interface [172.16.72.10]:
Configuring interface Async8:
    IPX network number [B]:
    Default client IP address for this interface [172.16.72.11]:
Configuring interface Async9:
    IPX network number [C]:
    Default client IP address for this interface [172.16.72.12]:
Configuring interface Async10:
    IPX network number [D]:
    Default client IP address for this interface [172.16.72.13]:
Configuring interface Async11:
    IPX network number [E]:
    Default client IP address for this interface [172.16.72.14]:
Configuring interface Async12:
    IPX network number [F]:
    Default client IP address for this interface [172.16.72.15]:
Configuring interface Async13:
    IPX network number [10]:
    Default client IP address for this interface [172.16.72.16]:
Configuring interface Async14:
    IPX network number [11]:
    Default client IP address for this interface [172.16.72.17]:
Configuring interface Async15:
    IPX network number [12]:
    Default client IP address for this interface [172.16.72.18]:
Configuring interface Async16:
    IPX network number [13]:
    Default client IP address for this interface [172.16.72.19]:
The following configuration command script was created:
hostname Router
enable secret 5 $1$krIg$emfYm/1OwHVspDuS8Gy0K1
enable password ww
line vty 0 4
password ww
snmp-server community public
!
no decnet routing
appletalk routing
ipx routing
ip routing
!
line 1 16
speed 57600
flowcontrol hardware
modem inout
!
arap network 20 ARA Dialins
line 1 16
arap enable
autoselect
!
! Turn off IPX to prevent network conflicts.
interface Ethernet0
no ipx network
interface Serial0
no ipx network
```

```
interface Serial1
no ipx network
!
interface Ethernet0
ip address 172.16.72.2 255.255.255.0
appletalk cable-range 1-1 1.204
appletalk zone Sales
ipx network 1
no mop enabled
!
interface Serial0
no shutdown
no ip address
ip unnumbered Ethernet0
appletalk cable-range 3-3
appletalk zone ZZ Serial
ipx network 3
no mop enabled
!
interface Serial1
no ip address
ip unnumbered Ethernet0
appletalk cable-range 2-2 2.2
appletalk zone ZZ Serial
ipx network 2
no mop enabled
!
Interface Async1
ipx network 4
ip unnumbered Ethernet0
peer default ip address 172.16.72.4
async mode interactive
!
Interface Async2
ipx network 5
ip unnumbered Ethernet0
peer default ip address 172.16.72.5
async mode interactive
!
Interface Async3
ipx network 6
ip unnumbered Ethernet0
peer default ip address 172.16.72.6
async mode interactive
!
Interface Async4
ipx network 7
ip unnumbered Ethernet0
peer default ip address 172.16.72.7
async mode interactive
async dynamic address
!
Interface Async5
ipx network 8
ip unnumbered Ethernet0
peer default ip address 172.16.72.8
async mode interactive
!
Interface Async6
ipx network 9
ip unnumbered Ethernet0
peer default ip address 172.16.72.9
async mode interactive
!
```

```
Interface Async7
ipx network A
ip unnumbered Ethernet0
peer default ip address 172.16.72.10
async mode interactive
!
Interface Async8
ipx network B
ip unnumbered Ethernet0
peer default ip address 172.16.72.11
async mode interactive
!
Interface Async9
ipx network C
ip unnumbered Ethernet0
peer default ip address 172.16.72.12
async mode interactive
!
Interface Async10
ipx network D
ip unnumbered Ethernet0
peer default ip address 172.16.72.13
async mode interactive
!
Interface Async11
ipx network E
ip unnumbered Ethernet0
peer default ip address 172.16.72.14
async mode interactive
!
Interface Async12
ipx network F
ip unnumbered Ethernet0
peer default ip address 172.16.72.15
async mode interactive
!
Interface Async13
ipx network 10
ip unnumbered Ethernet0
peer default ip address 172.16.72.16
async mode interactive
!
Interface Async14
ipx network 11
ip unnumbered Ethernet0
peer default ip address 172.16.72.17
async mode interactive
!
Interface Async15
ipx network 12
ip unnumbered Ethernet0
peer default ip address 172.16.72.18
async mode interactive
!
Interface Async16
ipx network 13
ip unnumbered Ethernet0
peer default ip address 172.16.72.19
async mode interactive
!
router igrp 15
network 172.16.0.0
!
end
```

```
Use this configuration? [yes/no]: yes
Building configuration...
Use the enabled mode 'configure' command to modify this configuration.


Router#
```

Related Commands

Command	Description
erase nvram:	Erases a file system.
show running-config	Displays the running configuration file. Command alias for the **more system:running-config** command.
show startup-config	Displays the startup configuration file. Command alias for the **more system:startup-config**command.
show version	Displays the configuration of the system hardware, the software version, the names and sources of configuration files, and the boot images.