Configuring Additional Router Features

This chapter contains instructions and information for entering basic configurations using the command-line interface (CLI).

Contents

This chapter contains the following sections:

Configuring the Domain Name and Domain Name Server

Configure a domain name and domain name server (DNS) for your router to make contacting other devices on your network more efficient. Use the following guidelines:

- To define a default domain name that the Cisco IOS XR software uses to complete unqualified hostnames (names without a dotted-decimal domain name), use the **domain-name** command in global configuration mode.
- To specify the address of one or more name servers to use for name and address resolution, use the **domain name-server** command in global configuration mode. If no name server address is specified, the default name server is 255.255.255.255 so the DNS lookup can be broadcast to the local network segment. If a DNS server is in the local network, it replies. If not, there might be a server that knows how to forward the DNS request to the correct DNS server.
- Use the **show hosts** command in EXEC mode to display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.

To configure the DNS and DNS server, complete the following steps:

SUMMARY STEPS

1. **configure**
2. **domain name** *domain-name-of-organization*
3. **domain name-server** *ipv4-address*
4. **commit**
5. **end**
6. **show hosts**

Examples

In the following example, the domain name and DNS are configured:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# domain name cisco.com
RP/0/RP0/CPU0:router(config)# domain name-server 10.1.1.1
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router(config)# end
RP/0/RP0/CPU0:router# show hosts

Default domain is cisco.com
Name/address lookup uses domain service
Name servers: 10.1.1.1
```

Related Documents

Related Topic	Document Title
Complete descriptions of the domain services commands	*Implementing Host Services and Applications on Cisco IOS XR Software* in the *Cisco IOS XR IP Addresses and Services Configuration Guide*

Configuring Telnet, HTTP, and XML Host Services

For security, some host services are disabled by default. Host services, such as Telnet, Extensible Markup Language (XML), and HTTP, can be optionally enabled using the commands described in this section. Host services provide the following features:

- Enabling the Telnet server allows users to log in to the router using IPv4 or IPv6 Telnet clients.
- Enabling the HTTP server allows users to log in to the router using the CWI.
- Enabling the XML agent enables XML Common Object Request Broker Architecture (CORBA) agent services so that you can manage and configure the router using an XML interface.

Note IPv6 is not supported on the Cisco XR 12000 Series Router.

Prerequisites

The following prerequisites must be met before configuring the Telnet, HTTP, and XML host services:

- For the XML and HTTP host services, the Manageability package must be installed and activated on the router.
- To enable the Secure Socket Layer (SSL) of the HTTP and XML services, the Security package must be installed and activated on the router.

See Chapter 5, "Managing Cisco IOS XR Software Packages," for information on installing and activating packages.

SUMMARY STEPS

1. **configure**
2. **telnet ipv4 server max-servers 5**
3. **telnet ipv6 server max-servers 5**
4. **http server**
5. **xml agent corba**
6. **commit**

Examples

In the following example, the host services are enabled:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# telnet ipv4 server max-servers 5
RP/0/RP0/CPU0:router(config)# telnet ipv6 server max-servers 5
RP/0/RP0/CPU0:router(config)# http server
RP/0/RP0/CPU0:router(config)# xml agent corba
RP/0/RP0/CPU0:router(config)# commit
```

Related Documents

Related Topic	Document Title
Installation and activation of the Manageability and Security Packages	Chapter 5, "Managing Cisco IOS XR Software Packages"
Descriptions of the HTTP and XML server commands	*Cisco IOS XR System Management Command Reference*
Descriptions of the Telnet commands	*Cisco IOS XR IP Addresses and Services Command Reference*

Managing Configuration History and Rollback

After each commit operation, a record of the committed configuration changes is saved. This record contains only the changes made during the configuration session; it does not contain the complete configuration. Each record is assigned a unique ID, known as a *commitID*.

When multiple commitIDs are present, you can use a commitID to identify a previous configuration to which you want to return, or you can use the commitID to load the configuration changes made during that configuration session. You can also load configuration changes from multiple commitIDs, and you can clear commitIDs. If you are thinking about rolling back the configuration to a specific commitID, consider the following guidelines:

- You cannot roll back to a configuration that was removed because of package incompatibility. Configuration rollbacks can succeed only when the configuration passes all compatibility checks with the currently active Cisco IOS XR software.
- If an incompatible configuration is found during the rollback operation, the operation fails and an error is displayed.

The Cisco IOS XR software automatically saves up to 100 of the most recent commitIDs. The following sections describe how to manage configuration changes and roll back to a previously committed configuration:

Displaying the CommitIDs

To display a history of up to 100 of the most recent commitIDs, enter the **show configuration commit list** command in EXEC mode. Up to 100 of the most recent commitIDs are saved by the system. Each commitID entry shows the user who committed configuration changes, the connection used to execute the commit, and commitID time stamp.

The commitIDs are shown in the “Label/ID” column. The following example shows the **show configuration commit list** command display:

```
RP/0/RP0/CPU0:router# show configuration commit list

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    1000000391 user_a     con0_33_1   CLI         19:29:18 UTC Wed Jan 10 2004
2    1000000390 user_a     con0_33_1   CLI         19:29:16 UTC Wed Jan 10 2004
3    1000000389 user_a     con0_33_1   CLI         19:29:15 UTC Wed Jan 10 2004
4    1000000388 user_a     con0_33_1   CLI         19:29:12 UTC Wed Jan 10 2004
5    1000000387 user_a     con0_33_1   CLI         19:26:16 UTC Wed Jan 10 2004
6    1000000386 user_a     con0_32_1   CLI         19:18:38 UTC Wed Jan 10 2004
7    1000000385 user_a     con0_33_1   CLI         19:14:09 UTC Wed Jan 10 2004
8    1000000384 user_a     con0_33_1   CLI         19:13:58 UTC Wed Jan 10 2004
9    1000000383 user_a     con0_33_1   CLI         19:13:33 UTC Wed Jan 10 2004
10   1000000382 user_a     con0_33_1   CLI         19:12:50 UTC Wed Jan 10 2004
11   1000000381 user_a     con0_33_1   CLI         19:12:48 UTC Wed Jan 10 2004
12   1000000380 user_a     con0_33_1   CLI         19:12:46 UTC Wed Jan 10 2004
13   1000000379 user_a     con0_33_1   CLI         19:12:43 UTC Wed Jan 10 2004
14   1000000378 user_a     con0_33_1   CLI         19:12:14 UTC Wed Jan 10 2004
15   1000000377 user_a     con0_33_1   CLI         19:10:47 UTC Wed Jan 10 2004
```

Displaying the Configuration History Log

To display the header records for up to 1000 commit events, enter the **show configuration commit history** command in EXEC mode, as shown in the following example.

The output from this command does not show the details of the entries, but allows you to display a larger list of the commit events that occurred. To display the commitIDs to which you can roll back, use the **show configuration commit list** command.

```
RP/0/RP0/CPU0:router# show configuration commit history

SNo. Label/ID    User      Line        Client      Time Stamp
~~~~ ~~~~~~~~    ~~~~      ~~~~        ~~~~~~      ~~~~~~~~~~
1    1000000144  user_a    vty0         CLI         00:16:51 UTC Thu May 11 2004
2    1000000143  user_a    vty0         CLI         00:04:32 UTC Thu May 11 2004
3    1000000142  user_a    0.0.0.0      XMLAgent    21:58:36 UTC Wed May 11 2004
4    1000000141  user_a    0.0.0.0      XMLAgent    21:46:07 UTC Wed May 11 2004
5    1000000140  user_b    con0_RP1_C   CLI         21:43:30 UTC Wed May 11 2004
6    1000000139  user_a    0.0.0.0      XMLAgent    21:40:13 UTC Wed May 11 2004
7    1000000138  user_a    0.0.0.0      XMLAgent    21:34:48 UTC Wed May 11 2004
8    1000000137  user_b    con0_RP1_C   CLI         21:32:10 UTC Wed May 11 2004
9    1000000136  user_a    0.0.0.0      XMLAgent    21:30:13 UTC Wed May 11 2004
10   1000000135  user_b     con0_RP1_C  CLI         19:45:04 UTC Wed May 11 2004
11   1000000134  user_b     con0_RP1_C  CLI         19:37:26 UTC Wed May 11 2004
12   1000000133  user_b     con0_RP1_C  CLI         19:36:27 UTC Wed May 11 2004
13   1000000132  user_b     con0_33_1   Rollback    18:34:45 UTC Wed May 11 2004
14   1000000131  user_b     con0_33_1   Rollback    18:32:37 UTC Wed May 11 2004
15   1000000130  user_b     con0_33_1   Rollback    18:31:09 UTC Wed May 11 2004
16   1000000129  user_b     con0_33_1   CLI         18:28:12 UTC Wed May 11 2004
17   1000000128  user_b     con0_33_1   CLI         18:27:22 UTC Wed May 11 2004
18   1000000127  user_b     con0_33_1   CLI         18:27:19 UTC Wed May 11 2004
19   1000000126  user_b     con0_33_1   Rollback    18:25:55 UTC Wed May 11 2004
20   1000000125  user_b     con0_33_1   Rollback    18:24:25 UTC Wed May 11 2004
```

Displaying the Configuration Changes Recorded in a CommitID

To display the configuration changes made during a specific commit session (commitID), enter the **show configuration commit changes** command followed by a commitID number, as shown in the following example:

```
RP/0/0/CPU0:router# show configuration commit changes  1000000071
Building configuration...
hostname router2
end
```

Previewing Rollback Configuration Changes

The **show configuration rollback changes** command allows you to preview the configuration changes that take place if you roll back the configuration to a specific commitID. For example, if you want to roll back the configuration to a specific point, all configuration changes made after that point must be undone. This rollback process is often accomplished by executing the “no” version of commands that must be undone.

To display the prospective rollback configuration changes from the current configuration to a specific session, enter the **show configuration rollback changes to** *commitId* command:

```
RP/0/RP0/CPU0:router# show configuration rollback changes to 1000000373

Building configuration...
interface Loopback2
no description
no ipv4 address 10.0.0.1 255.0.0.0
```

To display the prospective rollback configuration changes from the current configuration to a specified number of previous sessions, enter the **show configuration rollback changes last** *commit-range* command:

```
RP/0/RP0/CPU0:router# show configuration rollback changes last 2

Building configuration...
interface Loopback3
no description
no ipv4 address 10.0.1.1 255.0.0.0
exit
interface Loopback4
no description
no ipv4 address 10.0.0.1 255.0.0.0
end
```

In the preceding example, the command display shows the proposed rollback configuration changes for the last two commit IDs.

Rolling Back the Configuration to a Specific Rollback Point

When you roll back the configuration to a specific rollback point, you undo all configuration changes made during the session identified by the commit ID for that rollback point, and you undo all configuration changes made after that point. The rollback process rolls back the configuration and commits the rolled-back configuration.

Tip To preview the commands that undo the configuration during a rollback, use the **show configuration rollback changes** command.

To roll back the router configuration to a previously committed configuration, enter the **rollback configuration to** *commitId* command:

```
RP/0/RP0/CPU0:router# rollback configuration to 1000000325

Configuration successfully rolled back to '1000000325'.
```

Rolling Back the Configuration over a Specified Number of Commits

When you roll back the configuration over a specific number of commits, you do not have to enter a specific commit ID. Instead, you specify a number *x*, and the software undoes all configuration changes made in the last *x* committed configuration sessions. The rollback process rolls back the configuration and commits the rolled-back configuration.

Tip To preview the commands that undo the configuration during a rollback, use the **show configuration rollback changes** command.

To roll back to the last *x* commits made, enter the **rollback configuration last** *x* command; *x* is a number ranging from 1 to the number of saved commits in the commit database.

In the following example, a request is made to roll back the configuration changes made during the previous two commits:

```
RP/0/RP0/CPU0:router# rollback configuration last 2

Loading Rollback Changes.
Loaded Rollback Changes in 1 sec
Committing.
1 items committed in 1 sec (0)items/sec
Updating.
Updated Commit database in 1 sec
Configuration successfully rolled back 2 commits.
```

Loading the Configuration Changes for a Specific CommitID

You can load the configuration changes recorded by any commitID by entering the **load commit changes** command in global configuration mode. The **load commit changes** command must be followed by a commitID number, as shown in the following example:

```
RP/0/0/CPU0:router(config)# load commit changes 1000000066
Building configuration...
Loading.
21 bytes parsed in 1 sec (20)bytes/sec
```

The configuration changes are added to the target configuration and are not applied until you enter the **commit** command.

Tip To display the target configuration, enter the **show configuration** command.

Loading Rollback Configuration Changes to the Target Configuration

You can load rollback configuration changes to the target configuration by entering the **load rollback changes** command in global configuration mode. This command is similar to the **rollback configuration** command. The difference between the commands is that the **load rollback changes** command copies the rollback changes to the target configuration and does not commit the changes.

Tip To display the rollback changes, enter the **show configuration rollback changes** command.

To load rollback configuration changes from the current configuration to a specific session, enter the **load rollback changes to** *commitId* command:

```
RP/0/0/CPU0:router(config)# load rollback changes to 1000000068
Building configuration...
Loading.
233 bytes parsed in 1 sec (231)bytes/sec
```

To load rollback configuration changes from the current configuration to a specified number of previous sessions, enter the **load rollback changes last** *commit-range* command:

```
RP/0/0/CPU0:router(config)# load rollback changes last 6
Building configuration...
Loading.
221 bytes parsed in 1 sec (220)bytes/sec
```

In the preceding example, the command loads the rollback configuration changes for the last six commitIDs.

To load the rollback configuration for a specific commitID, enter the **load rollback changes** *commitId* command:

```
RP/0/0/CPU0:router(config)# load rollback changes 1000000060
Building configuration...
Loading.
199 bytes parsed in 1 sec (198)bytes/sec
```

Deleting CommitIDs

You can delete the oldest configuration commitIDs by entering the **clear configuration commit** command in EXEC mode. The **clear configuration commit** command must be followed by either the amount of disk space you want to reclaim or number of commitIDs you want to delete. To reclaim disk space from the oldest commitIDs, enter the **clear configuration commit** command followed by the keyword **diskspace** and number of kilobytes to reclaim:

```
RP/0/0/CPU0:router# clear configuration commit diskspace 50
Deleting 4 rollback points '1000000001' to '1000000004'
64 KB of disk space will be freed. Continue with deletion?[confirm]
```

To delete a specific number of the oldest commitIDs, enter the **clear configuration commit** command followed by the keyword **oldest** and number of commitIDs to delete:

```
RP/0/0/CPU0:router# clear configuration commit oldest 5
Deleting 5 rollback points '1000000005' to '1000000009'
80 KB of disk space will be freed. Continue with deletion?[confirm]
```

Saving and Loading Target Configuration Files

Target configurations can be saved to a separate file without committing them to the running configuration. Target configuration files can then be loaded at a later time and further modified or committed. The following sections describe how to save and load target configurations:

- Saving the Target Configuration to a File, page 3-10
- Loading the Target Configuration from a File, page 3-10

Saving the Target Configuration to a File

To save the configuration changes in the target configuration to a file, enter the **show configuration | file** *filename* command.

- If the full path of the file is not specified, the default directory for your account is used. You should always save your target configuration files to this location.
- The filename should end with the `.cfg` suffix for easy identification. This suffix is not required, but can help locate target configuration files. Example: `myconfig.cfg`

Tip If you have not changed directories since login, you can display your default directory by entering the **pwd** command.

In the following example, a target configuration file is saved to the root of disk0:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# show configuration | file disk0:myconfig.cfg
RP/0/RP0/CPU0:router(config)# abort
RP/0/RP0/CPU0:router#
```

Loading the Target Configuration from a File

Enter the **load** *filename* command to populate the target configuration with the contents of a previously saved configuration file. Consider the following when entering the *filename* argument:

- The *filename* argument specifies the configuration file to be loaded into the target configuration.
- If the full path of the file is not specified, the default location is used. You should always save your target configuration files to this location.

In the following example, a target configuration file is loaded into the current configuration session. The current configuration session is therefore populated with the contents of the file:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# load disk0:myconfig.cfg
RP/0/RP0/CPU0:router(config)# show configuration
Building configuration... interface POS 0/3/0/0   description My Pos Interface   ipv4
address 10.10.11.20 255.0.0.0
!end
```

Loading an Alternative Configuration at System Startup

When a router is reset or powered on, the last running configuration is loaded and used to operate the router.

You can also load an alternative configuration during system boot. See Appendix A, “Router Recovery and Management with ROM Monitor,” for information and instructions on this process.

Configuring Logging and Logging Correlation

System messages generated by the Cisco IOS XR software can be logged to a variety of locations based on the severity level of the messages. For example, you could direct information messages to the system console and also log debugging messages to a network server.

In addition, you can define correlation rules that group and summarize related events, generate complex queries for the list of logged events, and retrieve logging events through an XML interface.

The following sections describe logging and the basic commands used to log messages in Cisco IOS XR software:

Logging Locations and Severity Levels

Error messages can be logged to a variety of locations, as shown in Table 3-1.

Table 3-1 Logging Locations for System Error Messages

Logging Destination	Command (Global Configuration Mode)
console	**logging console**
vty terminal	**logging monitor**
external syslog server	**logging trap**
internal buffer	**logging buffered**

You can log messages based on the severity level of the messages, as shown in Table 3-2.

Table 3-2 Logging Severity Levels for System Error Messages

Level	Description
Level 0—Emergencies	System has become unusable.
Level 1—Alerts	Immediate action needed to restore system stability.
Level 2—Critical	Critical conditions that may require attention.
Level 3—Errors	Error conditions that may help track problems.
Level 4—Warnings	Warning conditions that are not severe.
Level 5—Notifications	Normal but significant conditions that bear notification.
Level 6—Informational	Informational messages that do not require action.
Level 7—Debugging	Debugging messages are for system troubleshooting only.

Alarm Logging Correlation

Alarm logging correlation is used to group and filter similar messages to reduce the amount of redundant logs and isolate the root causes of the messages.

For example, the original message describing a card online insertion and removal (OIR) and system state being up or down can be reported, and all subsequent messages reiterating the same event can be correlated. When you create correlation rules, a common root event that is generating larger volumes of follow-on error messages can be isolated and sent to the correlation buffer. An operator can extract all correlated messages for display later, should the need arise. See the *Cisco IOS XR System Management Configuration Guide* for more information.

Configuring Basic Message Logging

Numerous options for logging system messages in Cisco IOS XR software are available. This section provides a basic example.

To configure basic message logging, complete the following steps:

SUMMARY STEPS

1. **configure**
2. **logging** {*ip-address* | *hostname*}
3. **logging trap** *severity*
4. **logging console** [*severity*]
5. **logging buffered** [*severity* | *buffer-size*]
6. **commit**
7. **end**
8. **show logging**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**logging** {*ip-address* | *hostname*} **Example:** `RP/0/RP0/CPU0:router(config)# logging 10.1.1.1`	Specifies a syslog server host to use for system logging.
Step 3	**logging trap** *severity* **Example:** `RP/0/RP0/CPU0:router(config)# logging trap debugging`	Limits the logging of messages sent to syslog servers to only those messages at the specified level. • See Table 3-2 for a summary of the logging severity levels.

	Command or Action	Purpose
Step 4	**logging console** [*severity*] **Example:** `RP/0/RP0/CPU0:router(config)# logging console emergencies`	Logs messages on the console. • When a severity level is specified, only messages at that severity level are logged on the console. • See Table 3-2 for a summary of the logging severity levels.
Step 5	**logging buffered** [*severity* | *buffer-size*] **Example:** `RP/0/RP0/CPU0:router(config)# logging buffered 1000000`	Copies logging messages to an internal buffer. • Newer messages overwrite older messages after the buffer is filled. • Specifying a severity level causes messages at that level and numerically lower levels to be logged in an internal buffer. See Table 3-2 for a summary of the logging severity levels. • The buffer size is from 4096 to 4,294,967,295 bytes. Messages above the set limit are logged to the console.
Step 6	**commit** **Example:** `RP/0/RP0/CPU0:router(config)# commit`	Commits the target configuration to the router running configuration.
Step 7	**end** **Example:** `RP/0/RP0/CPU0:router(config)# end`	Ends the configuration session and returns to EXEC mode.
Step 8	**show logging** **Example:** `RP/0/RP0/CPU0:router# show logging`	Displays the messages that are logged in the buffer.

Examples

In the following example, basic message logging is configured:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# logging 10.1.1.1
RP/0/RP0/CPU0:router(config)# logging trap debugging
RP/0/RP0/CPU0:router(config)# logging console emergencies
RP/0/RP0/CPU0:router(config)# logging buffered 1000000
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router(config)# end
RP/0/RP0/CPU0:router# show logging
Syslog logging: enabled (10 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 103 messages logged
    Monitor logging: level debugging, 0 messages logged
    Trap logging: level informational, 0 messages logged
    Buffer logging: level debugging, 107 messages logged

Log Buffer (16384 bytes):
```

```
RP/0/RP0/CPU0:Apr  6 21:30:59.515 : alphadisplay[103][317]: alpha_display_drain_
queue: Draining 1 message from the queue of size = 1
RP/0/RP0/CPU0:Apr  6 21:31:03.099 : ingressq[227]: %INGRESSQ_DLL-3-HALF_DEPTH_PA
RT_DISCOVERED : ingressq dll: half depth memory detected, memory: DQ External QE
 Memory
--More--
```

Related Documents

For more information on message logging and configuration of alarm correlation, see the following Cisco documents:

Related Topic	Document Title
Configuration of system logging	*Cisco IOS XR System Management Configuration Guide*
Commands used to configure logging	*Cisco IOS XR System Management Command Reference*
Configuration of alarm correlation and generating complex queries	*Cisco IOS XR System Management Configuration Guide*
Commands used to configure alarm correlation	*Cisco IOS XR System Management Command Reference*
Retrieve logging events through an XML interface	*Cisco IOS XR XML API Guide*

Creating and Modifying User Accounts and User Groups

In the Cisco IOS XR software, users are assigned individual usernames and passwords. Each username is assigned to one or more user groups, each of which defines display and configuration commands the user is authorized to execute. This authorization is enabled by default in the Cisco IOS XR software, and each user must log in to the system using a unique username and password.

The following sections describe the basic commands used to configure users and user groups. For a summary of user accounts, user groups, and task IDs, see the "User Access Privileges" section on page 2-12.

Note The management of user accounts, user groups, and task IDs is part of the "AAA" feature in the Cisco IOS XR software. AAA stands for "authentication, authorization, and accounting," a suite of security features included in the Cisco IOS XR software. For more information on the AAA concepts and configuration tasks, see the *Cisco IOS XR System Security Configuration Guide* and the *Cisco IOS XR System Security Command Reference*. For instructions to activate software packages, see Chapter 5, "Managing Cisco IOS XR Software Packages."

Displaying Details About User Accounts, User Groups, and Task IDs

Table 3-3 summarizes the EXEC mode commands used to display details about user accounts, user groups, and task IDs.

Table 3-3 Commands to Display Details About Users and User Groups

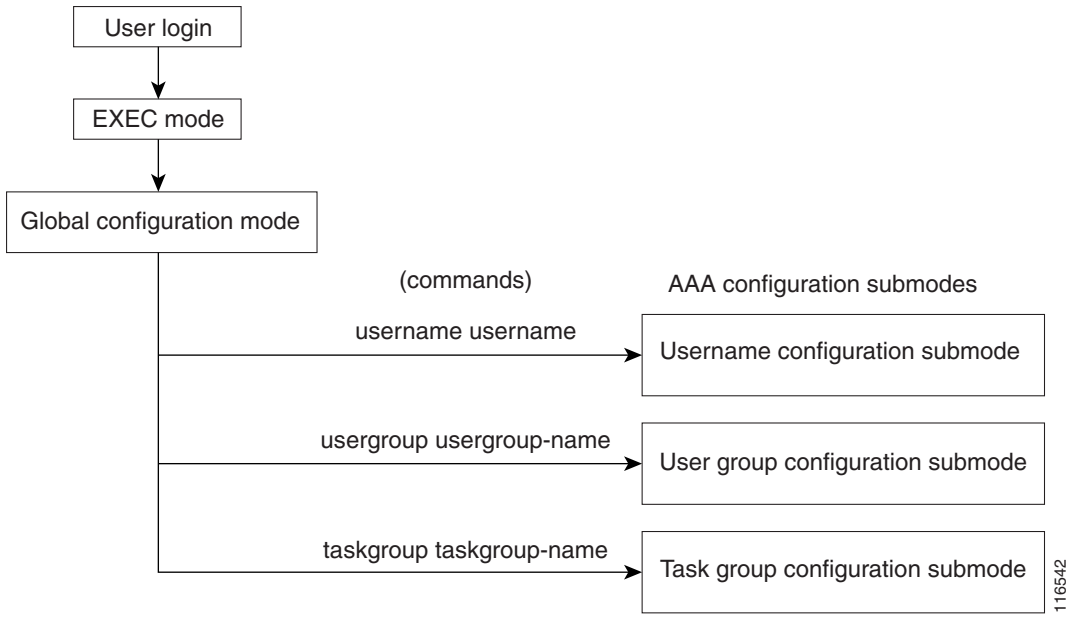
Command	Description
show aaa userdb *username*	Displays the task IDs and privileges assigned to a specific username. To display all users on the system, enter the command without a username.
show aaa usergroup *usergroup-name*	Displays the task IDs and privileges that belong to a user group. To display all groups on the system, enter the command without a group name.
show task supported	Displays all task IDs for the system. Only the root-system users, root-lr users, or users associated with the WRITE:AAA task ID can configure task groups.

Configuring User Accounts

User accounts, user groups, and task groups are created by entering the appropriate commands in one of the "AAA" configuration submodes, as shown in Figure 3-1.

This section describes the process to configure usernames. For instructions to configure user groups, task groups, and other AAA security features, see the *Cisco IOS XR System Security Configuration Guide.*

Figure 3-1 AAA Configuration Submodes



Creating Users and Assigning Groups

To create a user, assign a password, and assign the user to a group, perform the following procedure:

SUMMARY STEPS

1. **configure**
2. **username** *user-name*
3. **password** {**0** | **7**} *password*
   or
   secret {**0** | **5**} *password*
4. **group** *group-name*
5. Repeat Step 4 for each user group to be associated with the user specified in Step 2.
6. **end**
   or
   commit

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters global configuration mode.
Step 2	**username** *user-name* **Example:** `RP/0/RP0/CPU0:router(config)# username user1`	Creates a name for a new user (or identifies a current user) and enters username configuration submode. • The *user-name* argument can be only one word. Spaces and quotation marks are not allowed.

Step 3	**password** {**0** | **7**} *password*
	or
	secret {**0** | **5**} *password*
	
	Example:
	`RP/0/RP0/CPU0:router(config-un)# password 0 pwd1`
	or
	`RP/0/RP0/CPU0:router(config-un)# secret 5 pwd1`
		Specifies a password for the user named in Step 2.
		• Use the **secret** command to create a secure login password for the user names specified in Step 2.
		• Entering **0** following the **password** command specifies that an unencrypted (clear-text) password follows. Entering **7** following the **password** command specifies that an encrypted password follows.
		• Entering **0** following the **secret** command specifies that a secure unencrypted (clear-text) password follows. Entering **5** following the **secret** command specifies that a secure encrypted password follows.
		• Type **0** is the default for the **password** and **secret** commands.

	Command or Action	Purpose
Step 4	**group** *group-name* **Example:** `RP/0/RP0/CPU0:router(config-un)# group sysadmin`	Assigns the user named in Step 2 to a user group. • The user takes on all attributes of the user group, as defined by the user group association to various task groups. • Each user must be assigned to at least one user group. A user may belong to multiple user groups.
Step 5	Repeat Step 4 for each user group to be associated with the user specified in Step 2.	—

Step 6	**end**
	or
	commit
	
	Example:
	`RP/0/RP0/CPU0:router(config-un)# end`
	or
	`RP/0/RP0/CPU0:router(config-un)# commit`
		Saves configuration changes.
		• When you issue the **end** command, the system prompts you to commit changes:
		`Uncommitted changes found. Commit them?`
		– Entering **yes** saves configuration changes to the running configuration file, exits the configuration session, and returns the router to EXEC mode.
		– Entering **no** exits the configuration session and returns the router to EXEC mode without committing the configuration changes.
		• Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session.

Related Documents

For more information on configuration and management of users and user access privileges, see the following Cisco documents:

Related Topic	Document Title
Create users, assign users to user groups, create and modify user groups, and configure remote AAA access	*Cisco IOS XR System Security Configuration Guide*

Configuration Limiting

The Cisco IOS XR software places preset limits on the configurations you can apply to the running configuration of a router. These limits ensure that the router has sufficient system resources (such as RAM) for normal operations. Under most conditions, these preset limits are sufficient.

In some cases, for which a large number of configurations is required for a particular feature, it may be necessary to override the preset configuration limits. This override can be done only if configurations for another feature are low or unused. For example, if a router requires a large number of BGP configurations and Multiprotocol Label Switching (MPLS) is not being used at all, then the BGP limits can be increased to use the unused memory assigned to MPLS.

Caution Overriding the default configuration limits can result in a low-memory condition.

The following sections describe the limits you can configure, default and maximum values, and commands for configuring and displaying the configuration limits:

Static Route Configuration Limits

Table 3-4 summarizes the maximum limits for static routes, including the commands used to display and change the limits.

Table 3-4 Static Route Configuration Limits and Commands

Feature Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Global Configuration Mode)	Show Current Settings Command (EXEC Mode)
Maximum static IPv4 routes	4000	128,000	**route maximum ipv4** *n*	**show running-config route maximum**
Maximum static IPv6[1] routes	4000	128,000	**route maximum ipv6** *n*	**show running-config route maximum**

1. IPv6 is not supported on the Cisco XR 12000 Series Router.

Examples

In the following example, the maximum number of static IPv4 routes is changed to 5000. The new setting is then displayed with the **show running-config route maximum** command.

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(config)# route maximum ipv4 5000
RP/0/RP1/CPU0:router(config)# commit
RP/0/RP1/CPU0:Mar 30 15:50:38 : ipv4_static[214]: %IP_STATIC-6-CONFIG_MAXIMUM_CH
ANGE : The maximum number of configurations for static routes has been changed f
rom 4000 to 5000
RP/0/RP1/CPU0:Mar 30 15:50:39 : config[65740]: %LIBTARCFG-6-COMMIT : Configurati
on committed by user 'lab'.   Use 'show configuration commit changes 1000000538' to view
the c
hanges.
RP/0/RP1/CPU0:router(config)# end
RP/0/RP1/CPU0:Mar 30 15:50:46 : config[65740]: %SYS-5-CONFIG_I : Configured from
 console by lab

RP/0/RP1/CPU0:router# show running-config route maximum
route maximum ipv4 5000

RP/0/RP1/CPU0:router#
```

IS-IS Configuration Limits

Table 3-5 summarizes the maximum limits for IS-IS, including the commands used to display and change the limits.

Table 3-5 IS-IS Configuration Limits and Commands

Feature Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Address Family Configuration Mode)	Show Current Settings Command (EXEC Mode)
Maximum number of prefixes redistributed into IS-IS	10,000	28,000	**maximum-redistributed-prefixes** *n*	**show isis adjacency**
Number of active parallel paths for each route on a Cisco CRS-1 router	8	32	**maximum-paths** *n*	**show isis route**
Number of active parallel paths for each route on a Cisco XR 12000 Series Router	8	16	**maximum-paths** *n*	**show isis route**

Examples

In the following example, the maximum number of active parallel paths for each route is increased to 10, and the maximum number of prefixes redistributed into IS-IS is increased to 12,000:

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(config)# router isis 100 address-family ipv4
RP/0/RP1/CPU0:router(config-isis-af)# maximum-paths 10
RP/0/RP1/CPU0:router(config-isis-af)# maximum-redistributed-prefixes 12000
RP/0/RP1/CPU0:router(config-isis-af)# commit
RP/0/RP1/CPU0:Mar 30 14:11:07 : config[65739]: %LIBTARCFG-6-COMMIT : Configurati
on committed by user 'lab'.   Use 'show configuration commit changes 1000000535' to view
the c
hanges.
RP/0/RP1/CPU0:router(config-isis-af)#
```

OSPFv2 and v3 Configuration Limits

Table 3-6 summarizes the maximum limits for OSPF, including the commands used to display and change the limits.

Table 3-6 OSPFv2 and OSPFv3 Configuration Limits and Commands

Feature Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Router Configuration Mode)	Show Current Settings Command (EXEC Mode)
Maximum number of interfaces that can be configured for an OSPF instance	255	1024	**maximum interfaces** *n*	**show ospf**
Maximum routes redistributed into OSPF	10,000	28,672	**maximum redistributed-prefix** *n*	**show ospf** **Note** The maximum number of redistributed prefixes is displayed only if redistribution is configured.
Maximum number of parallel routes (maximum paths) on Cisco CRS-1s	32 (OSPFv2) 16 (OSPFv3)	32	**maximum paths** *n*	**show running-config router ospf** **Note** This command shows only changes to the default value. If the **maximum paths** command does not appear, the router is set to the default value.
Maximum number of parallel routes (maximum paths) on a Cisco XR 12000 Series Router	16	16	**maximum paths** *n*	**show running-config router ospf** **Note** This command shows only changes to the default value. If the **maximum paths** command does not appear, the router is set to the default value.

Examples

The following subsections provide the following examples:

- Maximum Interfaces for Each OSPF Instance, page 3-21
- Maximum Routes Redistributed into OSPF, page 3-22
- Number of Parallel Links (max-paths), page 3-22

Maximum Interfaces for Each OSPF Instance

In the following example, the **show ospf** command is used to display the maximum number of OSPF interfaces:

```
RP/0/RP1/CPU0:router# show ospf
 Routing Process "ospf 100" with ID 0.0.0.0
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 It is an area border router
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Initial LSA throttle delay 500 msecs
 Minimum hold time for LSA throttle 5000 msecs
 Maximum wait time for LSA throttle 5000 msecs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 Maximum number of configured interfaces 255
--More--
```

The following example configures the maximum interface limit on a router:

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(config)# router ospf 100
RP/0/RP1/CPU0:router(config-router)# maximum interfaces 600
RP/0/RP1/CPU0:router(config-router)# end
Uncommitted changes found, commit them? [yes]: y
RP/0/RP1/CPU0:Mar 30 16:12:39 : config[65740]: %LIBTARCFG-6-COMMIT : Configurati
on committed by user 'lab'.   Use 'show configuration commit changes 1000000540' to view
the c
hanges.
RP/0/RP1/CPU0:Mar 30 16:12:39 : config[65740]: %SYS-5-CONFIG_I : Configured from
 console by lab

RP/0/RP1/CPU0:router# show ospf

 Routing Process "ospf 100" with ID 0.0.0.0
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 It is an area border router
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Initial LSA throttle delay 500 msecs
 Minimum hold time for LSA throttle 5000 msecs
 Maximum wait time for LSA throttle 5000 msecs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 Maximum number of configured interfaces 600
--More--
```

Maximum Routes Redistributed into OSPF

In the following example, the **maximum redistributed-prefixes** command is used to set the maximum routes redistributed into OSPF:

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(config)# router ospf 100
RP/0/RP1/CPU0:router(config-router)# maximum redistributed-prefixes 12000
RP/0/RP1/CPU0:router(config-router)# end
Uncommitted changes found, commit them? [yes]: y
RP/0/RP1/CPU0:Mar 30 16:26:52 : config[65740]: %LIBTARCFG-6-COMMIT : Configurati
on committed by user 'lab'.   Use 'show configuration commit changes 1000000541' to view
the changes.
RP/0/RP1/CPU0:Mar 30 16:26:52 : config[65740]: %SYS-5-CONFIG_I : Configured from
 console by lab
RP/0/RP1/CPU0:router#
```

Number of Parallel Links (max-paths)

In the following example, the **maximum paths** command is used to set the maximum number of parallel routes:

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(config)# router ospf 100
RP/0/RP1/CPU0:router(config-router)# maximum paths 10
RP/0/RP1/CPU0:router(config-router)# end
Uncommitted changes found, commit them? [yes]: y
RP/0/RP1/CPU0:Mar 30 18:05:13 : config[65740]: %LIBTARCFG-6-COMMIT : Configurati
on committed by user 'lab'.   Use 'show configuration commit changes 1000000542' to view
the changes.
RP/0/RP1/CPU0:Mar 30 18:05:13 : config[65740]: %SYS-5-CONFIG_I : Configured from
 console by lab
RP/0/RP1/CPU0:router#
```

BGP Configuration Limits

The maximum number of BGP neighbors (peers) that can be configured is 1024. This number cannot be changed through configuration. Any attempt to configure additional neighbors beyond the limit fails.

To prevent neighbors (peers) from flooding BGP with advertisements, a limit is placed on the number of prefixes that can be accepted from a peer for each supported address family.

You can override the default limits for an address family with the **maximum-prefix** command. Table 3-7 summarizes the maximum configuration limits for BGP.

Table 3-7 BGP Configuration Limits and Commands

Feature Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Router Configuration Mode)	Show Current Settings Command (EXEC Mode)
Maximum number of neighbors (peers).	1024	1024	None. This limit cannot be changed or exceeded.	None
IPv4 unicast maximum prefixes that can be received from a neighbor	524,288	4,294,967,295	**maximum-prefix** *n*	**show bgp neighbor** *IP_address*

Table 3-7 BGP Configuration Limits and Commands (continued)

Feature Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Router Configuration Mode)	Show Current Settings Command (EXEC Mode)
IPv4 multicast maximum prefixes that can be received from a neighbor	131,072	4,294,967,295	**maximum-prefix** *n*	**show bgp neighbor** *IP_address*
IPv6[1] unicast maximum prefixes that can be received from a neighbor	131,072	4,294,967,295	**maximum-prefix** *n*	**show bgp neighbor** *IP_address*
Maximum equal-cost parallel routes to external peers	1	8	**maximum-paths** *n*	**show running-config** **Note** This command shows only changes to the default value. If the **maximum paths** command does not appear, the router is set to the default value.

1. IPv6 is not supported on the Cisco XR 12000 Series Router.

A cease-notification message is sent to the neighbor and the peering with the neighbor is terminated when the number of prefixes received from the peer for a given address family exceeds the maximum limit (either set by default or configured by the user) for that address family.

However, if the **warning-only** keyword (for the **maximum-prefix** command) is configured, the Cisco IOS XR software sends only a log message, but continues peering with the sender. If the peer is terminated, the peer stays down until the **clear bgp** command is issued.

The same set of actions (sending cease notification followed by the termination of the peering) is taken for a neighbor with which peering has already been established if you decide to configure a maximum that is less than the number of prefixes that have already been received from the neighbor.

Examples

The following example shows how to set the maximum number of IPv4 unicast prefixes allowed from the neighbor at 10.1.1.1 to 100,000:

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(config)# router bgp 100
RP/0/RP1/CPU0:router(config-bgp)# neighbor 10.1.1.1
RP/0/RP1/CPU0:router(config-bgp-nbr)# remote-as 1
RP/0/RP1/CPU0:router(config-bgp-nbr)# address-family ipv4 unicast
RP/0/RP1/CPU0:router(config-bgp-nbr-af)# maximum-prefix 100000
RP/0/RP1/CPU0:router(config-bgp-nbr-af)# commit
RP/0/RP1/CPU0:Mar 30 19:13:16 : config[65740]: %LIBTARCFG-6-COMMIT : Configurati
on committed by user 'lab'.   Use 'show configuration commit changes 1000000544' to view
the c
hanges.
RP/0/RP1/CPU0:Mar 30 19:13:17 : config[65740]: %SYS-5-CONFIG_I : Configured from
 console by lab
RP/0/RP1/CPU0:router(config-bgp-nbr-af)#
```

Routing Policy Language Line and Policy Limits

Two limits for Routing Policy Language (RPL) configurations exist:

1. Lines of configuration: The number of lines entered by the user, including the beginning and ending statements (that is “route-policy”). The lines of configuration for sets are also included.
2. Number of RPL policies: The number of policies that can be configured on the router. Policies are counted only once: Multiple use of the same policy counts as a single policy toward the limit 1.

The limits for RPL lines and policies are summarized in Table 3-8. You can change the default values up to the absolute maximum, but you cannot change the value to a number less than the number of items that are currently configured.

Table 3-8 Maximum Lines of RPL: Configuration Limits and Commands

Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Global Configuration Mode)	Show Current Settings Command (EXEC Mode)
Maximum number of RPL lines	65,536	131,072	**rpl maximum lines** *n*	**show rpl maximum lines**
Maximum number of RPL policies	3500	5000	**rpl maximum policies** *n*	**show rpl max policies**

Examples

In the following example, the **show rpl maximum** command is used in EXEC mode to display the current setting for RPL limits and number of each limit currently in use. A summary of the memory used by all of the defined policies is also shown below the limit settings.

```
RP/0/RP1/CPU0:router# show rpl maximum

                            Current    Current       Max
                             Total      Limit       Limit
-------------------------------------------------------------
Lines of configuration             0      65536     131072
Policies                           0       3500       5000
Compiled policies size (kB)        0
RP/0/RP1/CPU0:router#
```

In the next example, the **rpl maximum** command changes the currently configured line and policy limits. The **show rpl maximum** command displays the new settings.

```
RP/0/RP1/CPU0:router# configure
RP/0/RP1/CPU0:router(config)# rpl maximum policies 4000
RP/0/RP1/CPU0:router(config)# rpl maximum lines 80000
RP/0/RP1/CPU0:router(config)# commit
RP/0/RP1/CPU0:Apr  1 00:23:44.062 : config[65709]: %LIBTARCFG-6-COMMIT : Configu
ration committed by user 'UNKNOWN'.   Use 'show configuration commit changes 1000000010'
to vi
ew the changes.
RP/0/RP1/CPU0:router(config)# exit
RP/0/RP1/CPU0:Apr  1 00:23:47.781 : config[65709]: %SYS-5-CONFIG_I : Configured
from console by console
RP/0/RP1/CPU0:router# show rpl maximum

                            Current    Current       Max
                             Total      Limit       Limit
-------------------------------------------------------------
Lines of configuration             0      80000     131072
Policies                           0       4000       5000
Compiled policies size (kB)        0
RP/0/RP1/CPU0:router#
```

Multicast Configuration Limits

Table 3-9 summarizes the maximum limits for multicast configuration, including the commands used to display and change the limits.

Table 3-9 Multicast Configuration Limits and Commands

Feature Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Global Configuration Mode)	Show Current Settings Command (EXEC Mode)
Internet Group Management Protocol (IGMP) Limits				
Maximum number of groups used by IGMP and accepted by a router	50,000	75,000	**maximum groups** *n*	**show igmp summary**
Maximum number of groups for each interface accepted by a router	20,000	40,000	**maximum groups per-interface** *n*	**show igmp summary**
Multicast Source Discovery Protocol (MSDP) Limits				
Maximum MSDP Source Active (SA) entries	20,000	75,000	**maximum external-sa** *n*	**show msdp summary**
Maximum MSDP SA entries that can be learned from MSDP peers	20,000	75,000	**maximum peer-external-sa** *n*	**show msdp summary**
Protocol Independent Multicast (PIM) Limits				
Maximum PIM routes supported	100,000	200,000	**maximum routes** *n*	**show pim summary**
Maximum PIM egress states	300,000	600,000	**maximum route-interfaces** *n*	**show pim summary**
Maximum PIM registers	20,000	75,000	**maximum register-states** *n*	**show pim summary**
Maximum number of PIM group map ranges learned from Auto-RP	500	5000	**maximum group-mappings autorp** *n*	**show pim summary**

MPLS Configuration Limits

Table 3-10 summarizes the maximum limits for MPLS configuration, including the commands used to display and change the limits.

Table 3-10 MPLS Configuration Limits and Commands

Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Global Configuration Mode)	Show Current Settings Command (EXEC Mode)
Maximum traffic engineer (TE) tunnels head	2500	4000	**mpls traffic-eng maximum tunnels** *n*	**show mpls traffic-eng maximum tunnels**

Other Configuration Limits

Table 3-11 summarizes the maximum limits for additional configuration limits, including the commands used to display and change the limits.

Table 3-11 Additional Configuration Limits and Commands

Limit Description	Default Maximum Limit	Absolute Maximum Limit	Configuration Command (Global Configuration Mode)	Show Current Settings Command (EXEC Mode)
IPv4 ACL (access list and prefix list)	5000	9000	**ipv4 access-list oor acl threshold** *n*	**show ipv4 access-lists**
IPv4 ACE (access list and prefix list)	200,000	350,000	**ipv4 access-list oor ace threshold** *n*	**show ipv4 access-lists**
IPv6[1] ACL (access list and prefix list)	1000	2000	**ipv6 access-list oor acl threshold** *n*	**show ipv4 access-lists**
IPv6[1] ACE (access list and prefix list)	50,000	100,000	**ipv6 access-list oor ace threshold** *n*	**show ipv4 access-lists**

1. IPv6 is not supported on the Cisco XR 12000 Series Router.