System Security Command Reference for Cisco NCS 5000 Series Routers

First Published: 2015-12-23

Last Modified: 2022-11-30

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

CONTENTS

Preface

This preface contains these sections:

Changes to This Document

This table lists the technical changes made to this document since it was first released.

Table 1: Changes to This Document

Date	Summary
July 2022	Republished for Release 7.7.1
October 2021	Republished for Release 7.3.2
July 2021	Republished for Release 7.4.1
August 2020	Republished for Release 7.1.2
August 2020	Republished for Release 7.2.1
August 2019	Republished for Release 7.0.1
May 2019	Republished for Release 6.6.25
March 2019	Republished for Release 6.5.3.
January 2019	Republished for Release 6.5.2
December 2018	Republished for Release 6.6.1
August 2018	Republished for Release 6.5.1.
July 2018	Republished for Release 6.4.2
March 2018	Republished for Release 6.4.1
March 2018	Republished for Release 6.3.2

Date	Summary
September 2017	Republished for Release 6.3.1
July 2017	Republished for Release 6.2.2
March 2017	Republished for Release 6.2.1
August 2016	Republished for Release 6.1.2
December 2015	Initial release of this document.

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Authentication, Authorization, and Accounting Commands

This module describes the commands used to configure authentication, authorization, and accounting (AAA) services.

For detailed information about AAA concepts, configuration tasks, and examples, see the Configuring AAA Services chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Note Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

aaa accounting

To create a method list for accounting, use the **aaa accounting** command in the XR EXEC mode. To remove a list name from the system, use the **no** form of this command.

aaa accounting {**commands** | **exec** | **mobile** | **network** | **system** } {**default** | **list-name**} {**start-stop** | **stop-only**} {**none** | **method**}
no aaa accounting {**commands** | **exec** | **mobile** | **network**} {**default** | **list-name**}

Syntax Description

commands	Enables accounting for XR EXEC shell commands.
exec	Enables accounting of a XR EXEC session.
mobile	Enables Mobile IP related accounting events.
network	Enables accounting for all network-related service requests, such as Internet Key Exchange (IKE) and Point-to-Point Protocol (PPP).
system	Enables accounting for all system-related events.
event manager	Sets the authorization list for XR EXEC.
default	Uses the listed accounting methods that follow this keyword as the default list of methods for accounting services.
list-name	Character string used to name the accounting method list.
start-stop	Sends a "start accounting" notice at the beginning of a process and a "stop accounting" notice at the end of a process. The requested user process begins regardless of whether the "start accounting" notice was received by the accounting server.
stop-only	Sends a "stop accounting" notice at the end of the requested user process. Note: This is not supported with system accounting.
none	Uses no accounting.
method	Method used to enable AAA system accounting. The value is one of the following options: • **group tacacs+**—Uses the list of all TACACS+ servers for accounting. • **group radius**—Uses the list of all RADIUS servers for accounting. • **group** *named-group*—Uses a named subset of TACACS+ or RADIUS servers for accounting, as defined by the **aaa group server tacacs+** or **aaa group server radius** command.

Command Default

AAA accounting is disabled.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **aaa accounting** command to create default or named method lists defining specific accounting methods and that can be used on a per-line or per-interface basis. You can specify up to four methods in the method list. The list name can be applied to a line (console, aux, or vty template) to enable accounting on that particular line.

The Cisco IOS XR software supports both TACACS+ and RADIUS methods for accounting. The router reports user activity to the security server in the form of accounting records, which are stored on the security server.

Method lists for accounting define the way accounting is performed, enabling you to designate a particular security protocol that is used on specific lines or interfaces for particular types of accounting services.

For minimal accounting, include the **stop-only** keyword to send a "stop accounting" notice after the requested user process. For more accounting, you can include the **start-stop** keyword, so that TACACS+ or RADIUS sends a "start accounting" notice at the beginning of the requested process and a "stop accounting" notice after the process. The accounting record is stored only on the TACACS+ or RADIUS server.

The requested user process begins regardless of whether the "start accounting" notice was received by the accounting server.

Note This command cannot be used with TACACS or extended TACACS.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to define a default commands accounting method list, where accounting services are provided by a TACACS+ security server, with a stop-only restriction:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa accounting commands default stop-only group tacacs+
```

aaa accounting system default

To enable authentication, authorization, and accounting (AAA) system accounting, use the **aaa accounting system default** command in the XR Config mode. To disable system accounting, use the **no** form of this command.

aaa accounting system default {**start-stop** | **stop-only**} {**none** | **method**}
no aaa accounting system default

Syntax Description

start-stop	Sends a "start accounting" notice during system bootup and a "stop accounting" notice during system shutdown or reload.
stop-only	Sends a "stop accounting" notice during system shutdown or reload.
none	Uses no accounting.
method	Method used to enable AAA system accounting. The value is one of the following options: • **group tacacs+**—Uses the list of all TACACS+ servers for accounting. • **group radius**—Uses the list of all RADIUS servers for accounting. • **group** *named-group*—Uses a named subset of TACACS+ or RADIUS servers for accounting, as defined by the **aaa group server tacacs+** or **aaa group server radius** command.

Command Default

AAA accounting is disabled.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

System accounting does not use named accounting lists; you can define only the default list for system accounting.

The default method list is automatically applied to all interfaces or lines. If no default method list is defined, then no accounting takes place.

You can specify up to four methods in the method list.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to cause a "start accounting" record to be sent to a TACACS+ server when a router initially boots. A "stop accounting" record is also sent when a router is shut down or reloaded.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa accounting system default start-stop group tacacs+
```

aaa accounting update

To enable periodic interim accounting records to be sent to the accounting server, use the **aaa accounting update** command in the XR Config mode. To disable the interim accounting updates, use the **no** form of this command.

aaa accounting update {**periodic** *minutes*}
no aaa accounting update

Syntax Description

periodic *minutes*	(Optional) Sends an interim accounting record to the accounting server periodically, as defined by the *minutes* argument, which is an integer that specifies the number of minutes. The range is from 1 to 35791394 minutes.

Command Default

AAA accounting update is disabled.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

When used with the **periodic** keyword, interim accounting records are sent periodically as defined by the *minutes* argument. The interim accounting record contains all the accounting information recorded for that user up to the time the accounting record is sent.

Caution Using the **aaa accounting update** command with the **periodic** keyword can cause heavy congestion when many users are logged into the network.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to send periodic interim accounting records to the RADIUS server at 30-minute intervals:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa accounting update periodic 30
```

aaa authentication (XR-VM)

To create a method list for authentication, use the **aaa authentication** command in the XR Config mode or System Admin Config mode. To disable this authentication method, use the **no** form of this command.

aaa authentication {**login** | **ppp**} {**default***list-name*} *method-list*
no aaa authentication {**login** | **ppp**} {**default***list-name*} *method-list*

Syntax Description

login	Sets authentication for login.
ppp	Sets authentication for Point-to-Point Protocol.
default	Uses the listed authentication methods that follow this keyword as the default list of methods for authentication.
list-name	Character string used to name the authentication method list.
method-list	Method used to enable AAA system accounting. The value is one of the following options: • **group tacacs+**—Specifies a method list that uses the list of all configured TACACS+ servers for authentication. • **group radius**—Specifies a method list that uses the list of all configured RADIUS servers for authentication. • **group** *named-group* —Specifies a method list that uses a named subset of TACACS+ or RADIUS servers for authentication, as defined by the **aaa group server tacacs+** or **aaa group server radius** command. • **local**—Specifies a method list that uses the local username database method for authentication. AAA method rollover happens beyond the local method if username is not defined in the local group. • **line**—Specifies a method list that uses the line password for authentication.

Command Default

Default behavior applies the local authentication on all ports.

Command Modes

XR Config mode or System Admin Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **aaa authentication** command to create a series of authentication methods, or method list. You can specify up to four methods in the method list. A *method list* is a named list describing the authentication methods (such as TACACS+ or RADIUS) in sequence. The subsequent methods of authentication are used only if the initial method is not available, not if it fails.

The default method list is applied for all interfaces for authentication, except when a different named method list is explicitly specified—in which case the explicitly specified method list overrides the default list.

For console and vty access, if no authentication is configured, a default of local method is applied.

Note

- The **group tacacs+**, **group radius**, and **group** *group-name* forms of this command refer to a set of previously defined TACACS+ or RADIUS servers.
- Use the **tacacs-server host** or **radius-server host** command to configure the host servers.
- Use the **aaa group server tacacs+** or **aaa group server radius** command to create a named subset of servers.
- The **login** keyword, **local** option, and **group** option are available only in System Admin Config mode.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to specify the default method list for authentication, and also enable authentication for console in XR Config mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa authentication login default group tacacs+
```

The following example shows how to specify the remote method list for authentication, and also enable authentication for console in System Admin Config mode:

```
RP/0/RP0/CPU0:router# admin
sysadmin-vm:0_RP0# configure
sysadmin-vm:0_RP0(config)# aaa authentication users user lab
```

```
RP/0/RP0/CPU0:router#  admin
sysadmin-vm:0_RP0# configure
sysadmin-vm:0_RP0(config)# aaa authentication groups group aaa-r
```

aaa authorization (XR-VM)

To create a method list for authorization, use the **aaa authorization** command in the XR Config mode. To disable authorization for a function, use the **no** form of this command.

aaa authorization { **commands** | **eventmanager** | **exec** | **network** | **nacm** } { **default** *list-name* } { **none** | **local** | **prefer-external** | **only-external** | **group** { **tacacs** + | **radius** *group-name* } }
no aaa authorization { **commands** | **eventmanager** | **exec** | **network** | **nacm** } { **default** *list-name* }

Syntax Description

commands	Configures authorization for all XR EXEC mode shell commands.
eventmanager	Applies an authorization method for authorizing an event manager (fault manager).
exec	Configures authorization for an interactive (XR EXEC mode) session.
network	Configures authorization for network services, such as PPP or Internet Key Exchange (IKE).
nacm	Enables the nacm functionality.
default	Uses the listed authorization methods that follow this keyword as the default list of methods for authorization.
list-name	Character string used to name the list of authorization methods.
none	Uses no authorization. If you specify **none**, no subsequent authorization methods is attempted. However, the task ID authorization is always required and cannot be disabled.
local	Uses local authorization.
prefer-external	Adds the external group names to the list of local group names to determine the access control rules.
only-external	Uses the external group names to determine the access control rules.
group tacacs+	Uses the list of all configured TACACS+ servers for authorization.
group radius	Uses the list of all configured RADIUS servers for authorization. This method of authorization is not available for command authorization.
group *group-name*	Uses a named subset of TACACS+ or RADIUS servers for authorization as defined by the **aaa group server tacacs+** or **aaa group server radius** command.

Command Default

Authorization is disabled for all actions (equivalent to the method **none** keyword).

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.4.1	NACM **prefer-external** and **only-external** keywords are introduced.
Release 6.0	This command was introduced.

Usage Guidelines

Use the **aaa authorization** command to create method lists defining specific authorization methods that can be used on a per-line or per-interface basis. You can specify up to four methods in the method list.

Note The command authorization mentioned here applies to the one performed by an external AAA server and *not* for task-based authorization.

Method lists for authorization define the ways authorization will be performed and the sequence in which these methods will be performed. A method list is a named list describing the authorization methods (such as TACACS+), in sequence. Method lists enable you to designate one or more security protocols for authorization, thus ensuring a backup system in case the initial method fails. Cisco IOS XR software uses the first method listed to authorize users for specific network services; if that method fails to respond, Cisco IOS XR software selects the next method listed in the method list. This process continues until there is successful communication with a listed authorization method or until all methods defined have been exhausted.

Note Cisco IOS XR software attempts authorization with the next listed method only when there is no response (not a failure) from the previous method. If authorization fails at any point in this cycle—meaning that the security server or local username database responds by denying the user services—the authorization process stops and no other authorization methods are attempted.

The Cisco IOS XR software supports the following methods for authorization:

- **none—**The router does not request authorization information; authorization is not performed over this line or interface.
- **local—**Use the local database for authorization.
- **group tacacs+—**Use the list of all configured TACACS+ servers for authorization.
- **group radius—**Use the list of all configured RADIUS servers for authorization.
- **group group-name—**Uses a named subset of TACACS+ or RADIUS servers for authorization.

Method lists are specific to the type of authorization being requested. Cisco IOS XR software supports four types of AAA authorization:

- **Commands authorization—**Applies to the XR EXEC mode commands a user issues. Command authorization attempts authorization for all XR EXEC mode commands.

Note "Command" authorization is distinct from "task-based" authorization, which is based on the task profile established during authentication.

- XR EXEC mode **authorization—**Applies authorization for starting an XR EXEC mode session.

Note The **exec** keyword is no longer used to authorize the fault manager service. The **eventmanager** keyword (fault manager) is used to authorize the fault manager service. The **exec** keyword is used for EXEC authorization.

- **Network authorization—**Applies authorization for network services, such as IKE.
- **Event manager authorization—**Applies an authorization method for authorizing an event manager (fault manager). You are allowed to use TACACS+ or locald.

Note The **eventmanager** keyword (fault manager) replaces the **exec** keyword to authorize event managers (fault managers).

When you create a named method list, you are defining a particular list of authorization methods for the indicated authorization type. When defined, method lists must be applied to specific lines or interfaces before any of the defined methods are performed.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to define the network authorization method list named listname1, which specifies that TACACS+ authorization is used:

```
Router# configure
Router(config)# aaa authorization commands listname1 group tacacs+
Router(config)#commit
```

aaa authorization (System Admin-VM)

To create command rules and data rules on System Admin VM for user authorization, use the **aaa authorization** command in System Admin Config mode. To delete the command rules and data rules, use the **no** form of this command.

aaa authorization { **cmdrules cmdrule** { *integer* | **range** *integer* } [{ **action** *action-type* | **command** *cmd-name* | **context** *context-name* | **group** *group-name* | **ops** *ops-type* }] | **commands group** { **none** | **tacacs** } | **datarules datarule** { *integer* | **range** *integer* } [{ **action** *action-type* | **context** *context-name* | **group** *group-name* | **keypath** *keypath-name* | **namespace** *namespace-string* | **ops** *ops-type* }] }

Syntax Description

cmdrules	Configures command rules.
cmdrule *integer*	Specifies the command rule number.
range *integer*	Specifies the range of the command rules or data rules to be configured.
action	Specifies whether users are permitted or not allowed to perform the operation specified for the **ops** keyword.
action-type	Specifies the action type for the command rule or data rule. Available options are: **accept**, **accept_log** and **reject**.
command *cmd-name*	Specifies the command to which the command rule applies. The command must be entered within double-quotes. Example, **get**.
context *context-name*	Specifies to which type of connection the command rule or data rule applies. The connection type can be netconf, cli, or xml.
group *group-name*	Specifies the group to which the command rule or data rule applies. Example, **admin-r**.
ops *ops-type*	Specifies whether the user has read, execute, or read and execute permissions for the command. Available options for command rules are: **r**, rx, and **x**. To know the available options for data rules, use a **?** after the **ops** keyword.
commands group	Sets the command authorization lists for server groups. Available options are **none** that specifies no authorization and **tacacs** that specifies use of the list of all tacacs+ hosts.
datarules	Configures data rules.
datarule *integer*	Specifies the data rule number.
keypath	Specifies the keypath of the data element. If you enter an asterisk '*' for keypath, it indicates that the command rule is applicable to all configuration data.

namespace	Enter asterisk "*" to indicate that the data rule is applicable for all namespace values.

Command Default

None

Command Modes

System Admin Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

From Cisco IOS XR Software Release 7.4.1 and later, the system internally maps the users configured on the XR VM to System Admin VM of the router, based on the task table of the user on the XR VM. With this feature, NETCONF and gRPC users can access the admin-related information on the router even if their user profiles do not exist on System Admin VM. For a sample configuration, see the example section.

For more details, see the *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

This example shows how to create a command rule:

```
sysadmin-vm:0_RP0#config
sysadmin-vm:0_RP0(config)#aaa authorization cmdrules cmdrule 10 action accept command "show
 platform" context cli group group1 ops rx
```

This example shows how to create a data rule:

```
sysadmin-vm:0_RP0#config
sysadmin-vm:0_RP0(config)#aaa authorization datarules datarule 20 action accept context cli
 group group10 keypath * namespace * ops rwx
```

This example shows how to configure a command rule for a NETCONF or gRPC session to allow read access for **admin-r** group users:

```
sysadmin-vm:0_RP0(config)#aaa authorization cmdrules cmdrule 6 context netconf command get
 group admin-r ops rx action accept
```

show nacm (XR-VM)

To display information about NETCONF Access Control information such as users, groups, rule-lists and traces, use the **show nacm** command in XR Config mode. To disable authorization for a function, use the **no** form of this command.

show nacm {**summary** | | **users** [<*user-name*>] | | **groups** [<*group-name*>] | | **rule-list** [<*rule-list-name*> [| **rule** [<*rule-name*>]]] | | **trace**}

Syntax Description

summary	Displays NACM summary information.
Users	Displays list of users in NACM database.
user-name	Displays info for a given user-name.
groups	Displas list of groups in the NACM database.
group-name	Displays information for a given group name.
rule-list	Displays list of rule-lists in the NACM database.
rule-list-name	Displays info for given rule-list-name.
rule	Displays list of rules under the rule-list in the NACM database.
rule-name	Displays info for given rule-name under rule-name in the NACM database.
trace tacacs+	Displays NACM process traces.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.4.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
nacm	read

Examples

The following example shows how to use the show nacm command:

```
RP/0/RP0/CPU0:xr-nacm #show nacm summary
 NACM SUMMARY
```

```
 ------------------------------------------------------------------------------
 Enable Nacm : False
 Enable External Groups : True
 Number of Groups : 2
 Number of Users : 2
 Number of Rules : 2
 Number of Rulelist : 2
 Default Read : permit
 Default Write : permit
 Default Exec : permit
 Denied Operations : 0
 Denied Data Writes : 0
 Denied Notifications : 0
 ------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
RP/0/RP0/CPU0:xr-nacm#show nacm users
USERS LIST:
------------------------------------------------------------------------------
lab,    admin,
------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
RP/0/RP0/CPU0:xr-nacm#show nacm users lab

USER NAME: lab
------------------------------------------------------------------------------
Groups List For User:
root-lr,    root-system,
------------------------------------------------------------------------------

RP/0/RP0/CPU0:xr-nacm#


RP/0/RP0/CPU0:xr-nacm#show nacm groups

GROUPS LIST:
------------------------------------------------------------------------------
root-system,    root-lr,
------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
RP/0/RP0/CPU0:xr-nacm#show nacm groups root-system

GROUP NAME: root-system
------------------------------------------------------------------------------
Users List:
admin,    lab,
Rules List:
rule-list-1,    rule-list-2,
------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
RP/0/RP0/CPU0:xr-nacm#show nacm rule-list
RULELISTS:
------------------------------------------------------------------------------
   Rulelist Index       Rulelist Name
     rule-list-2        rule-list-2
     rule-list-1        rule-list-1
------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
RP/0/RP0/CPU0:xr-nacm#show nacm rule-list rule-list-1,rule-list-1
RULELIST NAME: rule-list-1
------------------------------------------------------------------------------
   Rule Index              Rule Name
           rule1        rule1
           rule2        rule2
   Group List
```

```
root-system,    root-lr,
-------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
RP/0/RP0/CPU0:xr-nacm#show nacm rule-list rule-list-1,rule-list-1 rule

Rule Info:
        Name:                   rule1
        Index:                  rule1
        Value:                  edit-config
        ModuleName:             *
        Action:                 permit
        RuleType:               Rpc
        Comment:
        AccessOperations:       All
        HitCount:               0
 -------------------------------------------------------------------------------
 Rule Info:
        Name:                   rule2
        Index:                  rule2
        Value:                  /nacm/rule-list
        ModuleName:             ietf-netconf-acm
        Action:                 deny
        RuleType:               Data
        Comment:
        AccessOperations:       Read,
        HitCount:               0
 -------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
RP/0/RP0/CPU0:xr-nacm#show nacm rule-list rule-list-1,rule-list-1 rule rule2,rule2
RULELIST NAME: rule-list-1
 -------------------------------------------------------------------------------
Rule Info:
        Name:                   rule2
        Index:                  rule2
        Value:                  /nacm/rule-list
        ModuleName:             ietf-netconf-acm
        Action:                 deny
        RuleType:               Data
        Comment:
        AccessOperations:       Read,
        HitCount:               0
 -------------------------------------------------------------------------------
RP/0/RP0/CPU0:xr-nacm#
```

Related Commands

Command	Description
aaa accounting, on page 3	Creates a method list for accounting.

aaa default-taskgroup

To specify a task group for both remote TACACS+ authentication and RADIUS authentication, use the **aaa default-taskgroup** command in the XR Config mode. To remove this default task group, enter the **no** form of this command.

aaa default-taskgroup *taskgroup-name*
no aaa default-taskgroup

Syntax Description

taskgroup-name	Name of an existing task group.

Command Default

No default task group is assigned for remote authentication.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **aaa default-taskgroup** command to specify an existing task group for remote TACACS+ authentication.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to specify taskgroup1 as the default task group for remote TACACS+ authentication:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa default-taskgroup taskgroup1
```

aaa group server radius

To group different RADIUS server hosts into distinct lists, use the **aaa group server radius** command in the XR Config mode. To remove a group server from the configuration list, enter the **no** form of this command.

aaa group server radius *group-name*
no aaa group server radius *group-name*

Syntax Description

group-name	Character string used to name the group of servers.

Command Default

This command is not enabled.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **aaa group server radius** command to group existing server hosts, which allows you to select a subset of the configured server hosts and use them for a particular service. A server group is used in conjunction with a global server-host list. The server group lists the IP addresses or hostnames of the selected server hosts.

Server groups can also include multiple host entries for the same server, as long as each entry has a unique identifier. The combination of an IP address and User Datagram Protocol (UDP) port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific authentication, authorization, and accounting (AAA) service. In other words, this unique identifier enables RADIUS requests to be sent to different UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service, for example, accounting, the second host entry acts as an automatic switchover backup to the first host entry. Using this example, if the first host entry fails to provide accounting services, the network access server tries the second host entry on the same device for accounting services. The RADIUS host entries are tried in the order in which they are configured in the server group.

All members of a server group must be the same type, that is, RADIUS.

The server group cannot be named radius or tacacs.

This command enters server group configuration mode. You can use the server command to associate a particular RADIUS server with the defined server group.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows the configuration of an AAA group server named radgroup1, which comprises three member servers:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa group server radius radgroup1
RP/0/RP0/CPU0:router(config-sg-radius)# server 10.0.0.5 auth-port 1700 acct-port 1701
RP/0/RP0/CPU0:router(config-sg-radius)# server 10.0.0.10 auth-port 1702 acct-port 1703
RP/0/RP0/CPU0:router(config-sg-radius)# server 10.0.0.20 auth-port 1705 acct-port 1706
```

Note If the **auth-port** *port-number* and **acct-port** *port-number* keywords and arguments are not specified, the default value of the *port-number* argument for the **auth-port** keyword is 1645 and the default value of the *port-number* argument for the **acct-port** keyword is 1646.

aaa group server tacacs+

To group different TACACS+ server hosts into distinct lists, use the **aaa group server tacacs+** command in the XR Config mode. To remove a server group from the configuration list, enter the **no** form of this command.

aaa group server tacacs+ *group-name*
no aaa group server tacacs+ *group-name*

Syntax Description

group-name	Character string used to name a group of servers.

Command Default

This command is not enabled.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The AAA server-group feature introduces a way to group existing server hosts. The feature enables you to select a subset of the configured server hosts and use them for a particular service.

The **aaa group server tacacs+** command enters server group configuration mode. The **server** command associates a particular TACACS+ server with the defined server group.

A *server group* is a list of server hosts of a particular type. The supported server host type is TACACS+ server hosts. A server group is used with a global server host list. The server group lists the IP addresses or hostnames of the selected server hosts.

The server group cannot be named radius or tacacs.

Note Group name methods refer to a set of previously defined TACACS+ servers. Use the **tacacs-server host** command to configure the host servers.

From Cisco IOS XR Software Release 7.4.1 and later, you can configure a hold-down timer for TACACS+ server. For details, see the **holddown-time** command.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows the configuration of an AAA group server named tacgroup1, which comprises three member servers:

```
RP/0/RP0/CPU0:router# configure
```

```
RP/0/RP0/CPU0:router(config)# aaa group server tacacs+ tacgroup1
RP/0/RP0/CPU0:router(config-sg-tacacs)# server 192.168.200.226
RP/0/RP0/CPU0:router(config-sg-tacacs)# server 192.168.200.227
RP/0/RP0/CPU0:router(config-sg-tacacs)# server 192.168.200.228
```

aaa password-policy

To define a AAA password security policy, use the **aaa password-policy** command in XR Config mode. To remove the AAA password security policy, use the **no** form of this command.

aaa password-policy *policy-name* { **authen-max-attempts** *authen-max-attempts* | **lifetime** { **years** | **months** | **days** | **hours** | **minutes** | **seconds** } *lifetime* | **lockout-time** { **days** | **hours** | **minutes** | **seconds** } *lockout-time* | **lower-case** *lower-case* | **max-length** *max-length* | **min-char-change** *min-char-change* | **min-length** *min-length* | **numeric** *numeric* | **special-char** *special-char* | **upper-case** *upper-case* }

Syntax Description

policy-name	Specifies the name of the password, in characters.
authen-max-attempts	Specifies, in integer, the maximum number of authentication failure attempts allowed for a user, in order to restrict users who authenticate with invalid login credentials.
lifetime	Specifies the maximum lifetime for the password, the value of which is specified in integer, as years, months, days, hours, minutes or seconds.
lockout-time	Specifies, in integer, the duration (in days, hours, minutes or seconds) for which the user is locked out when he exceeds the maximum limit of authentication failure attempts allowed.
lower-case	Specifies the number of lower case alphabets allowed in the password policy, in integer.
max-length	Specifies the maximum length of the password, in integer.
min-char-change	Specifies the number of character change required between subsequent passwords, in integer.
min-length	Specifies the maximum length of the password, in integer.
numeric	Specifies the number of numerals allowed in the password policy, in integer.
special-char	Specifies the number of special characters allowed in the password policy, in integer.
upper-case	Specifies the number of upper case alphabets allowed in the password policy, in integer.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.2.1	The command options (except a few mentioned in the usage guidelines section) were extended to user secret as well.
Release 6.2.1	This command was introduced.

Usage Guidelines

AAA password security policy works as such for Cisco IOS XR platforms. Whereas, this feature is supported only on XR VM, for Cisco IOS XR 64 bit platforms and Cisco NCS 5000 Series Routers.

For more details on the usage of each option of this command, refer the section on *AAA Password Security for FIPS Compliance* in *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

You must configure both **authen-max-attempts** and **lockout-time** in order for the lock out functionality to take effect.

The **min-char-change** option is effective only for password change through logon, and not for password change by configuration.

Use **username** command along with **password-policy** option, in the XR Config mode, to associate the password policy with a particular user.

From Cisco IOS XR Software Release 7.2.1 and later, most of the options of the **aaa password-policy** command listed in the syntax above are applicable to user password as well as secret. Whereas, the options listed below are supported only for password, and not for secret:

- **max-char-repetition**
- **min-char-change**
- **restrict-password-reverse**
- **restrict-password-advanced**

Among the NCS540 router variants, the **restrict-consecutive-characters** option is applicable only for the following variants:

- N540-28Z4C-SYS-A/D
- N540X-16Z4G8Q2C-A/D
- N540-12Z20G-SYS-A/D
- N540X-12Z16G-SYS-A/D

This table lists the default, maximum and minimum values of various command variables:

Command Variables	Default Value	Maximum Value	Minimum Value
policy-name	None	253	1
max-length	253	253	2
min-length	2	253	2
special-char	0	253	0
upper-case	0	253	0
lower-case	0	253	0
numeric	0	253	0

Command Variables	Default Value	Maximum Value	Minimum Value
For **lifetime** :	0	99	1
years	0	11	1
months	0	30	1
days	0	23	1
hours	0	59	1
minutes	0	59	1
seconds			
min-char-change	4	253	0
authen-max-attempts	0	24	1
For **lockout-time** :	0	255	1
days	0	23	1
hours	0	59	1
minutes	0	59	1
seconds			

Task ID

Task ID	Operation
aaa	read, write

This example shows how to define a AAA password security policy:

```
RP/0/RP0/CPU0:router(config)#aaa password-policy test-policy
RP/0/RP0/CPU0:router(config-aaa)#min-length 8
RP/0/RP0/CPU0:router(config-aaa)#max-length 15
RP/0/RP0/CPU0:router(config-aaa)#lifetime months 3
RP/0/RP0/CPU0:router(config-aaa)#min-char-change 5
RP/0/RP0/CPU0:router(config-aaa)#authen-max-attempts 3
RP/0/RP0/CPU0:router(config-aaa)#lockout-time days 1
```

Related Commands

Command	Description
show aaa password-policy	Displays the details of AAA password policy.
username, on page 108	

accounting (line)

To enable authentication, authorization, and accounting (AAA) accounting services for a specific line or group of lines, use the **accounting** command. To disable AAA accounting services, use the **no** form of this command.

accounting {**commands** | **exec**} {**default***list-name*}
no accounting {**commands** | **exec**}

Syntax Description

commands	Enables accounting on the selected lines for all XR EXEC mode shell commands.
exec	Enables accounting of XR EXEC mode session.
default	The name of the default method list, created with the **aaa accounting** command.
list-name	Specifies the name of a list of accounting methods to use. The list is created with the **aaa accounting** command.

Command Default

Accounting is disabled.

Command Modes

Line template configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

After you enable the **aaa accounting** command and define a named accounting method list (or use the default method list) for a particular type of accounting, you must apply the defined lists to the appropriate lines for accounting services to take place. Use the **accounting** command to apply the specified method lists to the selected line or group of lines. If a method list is not specified this way, no accounting is applied to the selected line or group of lines.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to enable command accounting services using the accounting method list named *listname2* on a line template named *configure:*

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# line template configure
RP/0/RP0/CPU0:router(config-line)# accounting commands listname2
```

authorization (line)

To enable authentication, authorization, and accounting (AAA) authorization for a specific line or group of lines, use the **authorization** command in line template configuration mode. To disable authorization, use the **no** form of this command.

authorization {**commands** | **exec** | **eventmanager**} {**default***list-name*}
no authorization {**commands** | **exec** | **eventmanager**}

Syntax Description

commands	Enables authorization on the selected lines for all commands.
exec	Enables authorization for an interactive XR EXEC mode session.
default	Applies the default method list, created with the **aaa authorization** command.
eventmanager	Sets eventmanager authorization method. This method is used for the embedded event manager.
list-name	Specifies the name of a list of authorization methods to use. If no list name is specified, the system uses the default. The list is created with the **aaa authorization** command.

Command Default

Authorization is not enabled.

Command Modes

Line template configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

After you use the **aaa authorization** command to define a named authorization method list (or use the default method list) for a particular type of authorization, you must apply the defined lists to the appropriate lines for authorization to take place. Use the **authorization** command to apply the specified method lists (or, if none is specified, the default method list) to the selected line or group of lines.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to enable command authorization using the method list named *listname4* on a line template named *configure:*

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# line template configure
RP/0/RP0/CPU0:router(config-line)# authorization commands listname4
```

description (AAA)

To create a description of a task group or user group during configuration, use the **description** command in task group configuration or user group configuration mode. To delete a task group description or user group description, use the **no** form of this command.

description *string*
no description

Syntax Description

string	Character string describing the task group or user group.

Command Default

None

Command Modes

Task group configuration

User group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **description** command inside the task or user group configuration submode to define a description for the task or user group, respectively.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows the creation of a task group description:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# taskgroup alpha
RP/0/RP0/CPU0:router(config-tg)# description this is a sample taskgroup
```

The following example shows the creation of a user group description:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# usergroup alpha
RP/0/RP0/CPU0:router(config-ug)# description this is a sample user group
```

group (AAA)

To add a user to a group, use the **group** command in username configuration mode. To remove the user from a group, use the **no** form of this command.

group {**cisco-support** | **maintenance** | **netadmin** | **operator** | **provisioning** | **retrieve** | **root-lr** | **serviceadmin** | **sysadmin***group-name*}
no group {**cisco-support** | **maintenance** | **netadmin** | **operator** | **provisioning** | **retrieve** | **root-lr** | **serviceadmin** | **sysadmin***group-name*}

Syntax Description

cisco-support	Adds the user to the predefined Cisco support personnel group. **Note** Starting from IOS XR 6.0 release, the cisco-support group is combined with the root-system group. This means a user who is part of the root-system group can also access commands that are included in the cisco-support group.
maintenance	Adds the user to the predefined SCAPA maintenance group.
netadmin	Adds the user to the predefined network administrators group.
operator	Adds the user to the predefined operator group.
provisioning	Adds the user to the predefined SCAPA provisioning group.
retrieve	Adds the user to the predefined SCAPA retrieve group.
root-lr	Adds the user to the predefined root-lr group. Only users with root-lr authority may use this option.
serviceadmin	Adds the user to the predefined service administrators group.
sysadmin	Adds the user to the predefined system administrators group.
group-name	Adds the user to a named user group that has already been defined with the **usergroup** command.

Command Default

None

Command Modes

Username configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **group** command in username configuration mode. To access username configuration mode, use the username, on page 108 command in XR Config mode.

If the **group** command is used in System Admin Config mode, only cisco-support keywords can be specified.

The privileges associated with the cisco-support group are now included in the root-system group. The cisco-support group is no longer required to be used for configuration.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to assign the user group operator to the user named user1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# username user1
RP/0/RP0/CPU0:router(config-un)# group operator
```

holddown-time (TACACS+)

To specify a duration for which an unresponsive TACACS+ server is to be marked as down, and not be used for sending further client requests for that duration, use the **holddown-time** command in various configuration modes. To disable this feature, use the **no** form of this command or configure the hold down timer value as zero.

holddown-time *time*

Syntax Description

time	Specifies the hold-down timer value, in seconds. The range is from 0 to 1200. Zero indicates that the hold-down timer feature is disabled.

Command Default

By default, the TACACS+ hold-down timer is disabled.

Command Modes

TACACS server

TACACS+ server group

TACACS+ private server

Command History

Release	Modification
Release 7.4.1	This command was introduced.

Usage Guidelines

Note To set the hold-down timer at global level, use the **tacacs-server holddown-time** command in XR Config mode.

While selecting the timer at various configuration levels, the system gives preference to the one which is more specific to the server. That is, the server-level timer has the highest precedence, followed by server group-level and finally, the global-level.

Also, see the *Guidelines for Configuring Hold-Down Timer for TACACS+* section in the *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to mark an unresponsive TACACS+ server as being down, and not to use it for sending further client requests for a duration of 35 seconds:

```
Router(config)#tacacs-server host 10.105.236.102 port 2020
Router(config-tacacs-host)#holddown-time 35
```

This example shows how to set a hold-down timer at global level:

```
Router#configure
Router(config)#tacacs-server holddown-time 30
```

This example shows how to set a hold-down timer at server-group level:

```
Router#configure
Router(config)#aaa group server tacacs+ test-group
Router(config-sg-tacacs)#holddown-time 40
```

This example shows how to set a hold-down timer at private server level:

```
Router(config)#aaa group server tacacs+ test-group
Router(config-sg-tacacs)#server-private 10.105.236.109 port 2020
Router(config-sg-tacacs-private)#holddown-time 55
Router(config-sg-tacacs-private)#commit
```

Related Commands

Command	Description
aaa group server tacacs+, on page 21	Groups different TACACS+ server hosts into distinct lists.
server-private (TACACS+), on page 59	Configures the IP address of the private TACACS+ server for the group server.
tacacs-server host, on page 92	Configures a TACACS+ host server.

inherit taskgroup

To enable a task group to derive permissions from another task group, use the **inherit taskgroup** command in task group configuration mode.

inherit taskgroup {*taskgroup-name* | **netadmin** | **operator** | **sysadmin** | **cisco-support** | **root-lr** | **serviceadmin**}

Syntax Description

taskgroup-name	Name of the task group from which permissions are inherited.
netadmin	Inherits permissions from the network administrator task group.
operator	Inherits permissions from the operator task group.
sysadmin	Inherits permissions from the system administrator task group.
cisco-support	Inherits permissions from the cisco support task group.
root-lr	Inherits permissions from the root-lr task group.
serviceadmin	Inherits permissions from the service administrators task group.

Command Default

None

Command Modes

Task group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **inherit taskgroup** command to inherit the permissions (task IDs) from one task group into another task group. Any changes made to the taskgroup from which they are inherited are reflected immediately in the group from which they are inherited.

Task ID

Task ID	Operations
aaa	read, write

Examples

In the following example, the permissions of task group tg2 are inherited by task group tg1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# taskgroup tg1
RP/0/RP0/CPU0:router(config-tg)# inherit taskgroup tg2
RP/0/RP0/CPU0:router(config-tg)# end
```

inherit usergroup

To enable a user group to derive characteristics of another user group, use the **inherit usergroup** command in user group configuration mode.

inherit usergroup *usergroup-name*

Syntax Description

usergroup-name	Name of the user group from which permissions are to be inherited.

Command Default

None

Command Modes

User group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Each user group is associated with a set of task groups applicable to the users in that group. A task group is defined by a collection of task IDs. Task groups contain task ID lists for each class of action. The task permissions for a user are derived (at the start of the EXEC or XML session) from the task groups associated with the user groups to which that user belongs.

User groups support inheritance from other user groups. Use the **inherit usergroup** command to copy permissions (task ID attributes) from one user group to another user group. The "destination" user group inherits the properties of the inherited group and forms a union of all task IDs specified in those groups. For example, when user group A inherits user group B, the task map of the user group A is a union of that of A and B. Cyclic inclusions are detected and rejected. User groups cannot inherit properties from predefined groups, such as root-system users, root-sdr users, netadmin users, and so on. Any changes made to the usergroup from which it is inherited are reflected immediately in the group from which it is inherited.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to enable the purchasing user group to inherit properties from the sales user group:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# usergroup purchasing
RP/0/RP0/CPU0:router(config-ug)# inherit usergroup sales
```

key (TACACS+)

To specify an authentication and encryption key shared between the AAA server and the TACACS+ server, use the **key (TACACS+)** command in TACACS host configuration mode. To disable this feature, use the **no** form of this command.

key {**0** *clear-text-key* | **7** *encrypted-keyauth-key*}
no key {**0** *clear-text-key* | **7** *encrypted-keyauth-key*}

Syntax Description

0 *clear-text-key*	Specifies an unencrypted (cleartext) shared key.
7 *encrypted-key*	Specifies an encrypted shared key.
auth-key	Specifies the unencrypted key between the AAA server and the TACACS+ server.

Command Default

None

Command Modes

TACACS host configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The TACACS+ packets are encrypted using the key, and it must match the key used by the TACACS+ daemon. Specifying this key overrides the key set by the **tacacs-server key** command for this server only.

The key is used to encrypt the packets that are going from TACACS+, and it should match with the key configured on the external TACACS+ server so that the packets are decrypted properly. If a mismatch occurs, the result fails.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to set the encrypted key to anykey

```
RP/0/RP0/CPU0:router(config)# tacacs-server host 209.165.200.226
RP/0/RP0/CPU0:router(config-tacacs-host)# key anykey
```

login authentication

To enable authentication, authorization, and accounting (AAA) authentication for logins, use the **login authentication** command in line template configuration mode. To return to the default authentication settings, use the **no** form of this command.

login authentication {**default***list-name*}
no login authentication

Syntax Description

default	Default list of AAA authentication methods, as set by the **aaa authentication login** command.
list-name	Name of the method list used for authenticating. You specify this list with the **aaa authentication login** command.

Command Default

This command uses the default set with the **aaa authentication login** command.

Command Modes

Line template configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The **login authentication** command is a per-line command used with AAA that specifies the name of a list of AAA authentication methods to try at login.

Caution If you use a *list-name* value that was not configured with the **aaa authentication login** command, the configuration is rejected.

Entering the **no** form of the **login authentication** command has the same effect as entering the command with the **default** keyword.

Before issuing this command, create a list of authentication processes by using the **aaa authentication login** command.

Task ID

Task ID	Operations
aaa	read, write
tty-access	read, write

Examples

The following example shows that the default AAA authentication is used for the line template *template1*:

```
RP/0/RP0/CPU0:router# configure
```

```
RP/0/RP0/CPU0:router(config)# line template template1
RP/0/RP0/CPU0:router(config-line)# login authentication default
```

The following example shows that the AAA authentication list called *list1* is used for the line template *template2*:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# line template template2
RP/0/RP0/CPU0:router(config-line)# login authentication list1
```

nacm enable-external-policies

To enable dynamic NETCONF Access Control Model (NACM) policy authorization on a router, use the **nacm enable-external-policies** command in the XR Config mode. To remove the configuration, use the **no** form of this command.

nacm enable-external-policies

Syntax Description

This command has no keywords or arguments.

Command Default

Disabled, by default.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.8.1	This command was introduced.

Usage Guidelines

If this configuration is not present, update the NACM policies manually on each router.

Task ID

Task ID	Operation
nacm	read, write

This example shows how to enable the dynamic NACM on a router.

```
Router#configure
Router(config)# nacm enable-external-policies
Router(config)# commit
```

password (AAA)

To create a login password for a user, use the **password** command in username configuration mode or line template configuration mode. To remove the password, use the **no** form of this command.

password {[**0**] | **7** *password*}
no password {**0** | **7** *password*}

Syntax Description

0	(Optional) Specifies that an unencrypted clear-text password follows.
7	Specifies that an encrypted password follows.
password	Specifies the unencrypted password text to be entered by the user to log in, for example, “lab”. If encryption is configured, the password is not visible to the user. Can be up to 253 characters in length.

Command Default

The password is in unencrypted clear text.

Command Modes

Username configuration

Line template configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

You can specify one of two types of passwords: encrypted or clear text.

When an XR EXEC modeprocess is started on a line that has password protection, the process prompts for the password. If the user enters the correct password, the process issues the prompt. The user can try three times to enter a password before the process exits and returns the terminal to the idle state.

Passwords are two-way encrypted and should be used for applications such as PPP that need decryptable passwords that can be decrypted.

Note The **show running-config** command always displays the clear-text login password in encrypted form when the **0** option is used.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to establish the unencrypted password *pwd1* for user. The output from the **show** command displays the password in its encrypted form.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# username user1
RP/0/RP0/CPU0:router(config-un)# password 0 pwd1
RP/0/RP0/CPU0:router(config-un)# commit
RP/0/RP0/CPU0:router(config-un)# show running-config
Building configuration...
username user1
 password 7 141B1309
```

policy (AAA)

To configure a policy that is common for user password as well as secret, use the **policy** command in username configuration mode. To remove this configuration, use the **no** form of this command.

policy *policy-name*

Syntax Description

policy-name	Specifies the name of the policy that is common for user password as well as secret.

Command Default

None

Command Modes

username

Command History

Release	Modification
Release 7.2.1	This command was introduced.

Usage Guidelines

For detailed usage guidelines for this command, see the *Guidelines to Configure Password Policy for User Secret* section in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Task ID

Task ID	Operation
aaa	read, write

This example shows how to configure a password policy that applies to both the password and the secret of the user.

```
Router#configure
Router(config)#username user1
Router(config-un)#policy test-policy1
Router(config-un)#secret 10
$6$dmwuW0Ajicf98W0.$y/vzynWF1/OcGxwBwHs79VAy5ZZLhoHd7TicR4mOo8IIVriYCGAKW0A.w1JvTPO7IbZry.DxHrE3SN2BBzBJe0
Router(config-un)#commit
```

Related Commands

Command	Description
username, on page 108	

radius-server dead-criteria time

To specify the minimum amount of time, in seconds, that must elapse from the time that the router last received a valid packet from the RADIUS server to the time the server is marked as dead, use the **radius-server dead-criteria time** command in XR Config mode. To disable the criteria that were set, use the **no** form of this command.

radius-server dead-criteria time *seconds*
no radius-server dead-criteria time *seconds*

Syntax Description

seconds	Length of time, in seconds. The range is from 1 to120 seconds. If the *seconds* argument is not configured, the number of seconds ranges from 10 to 60, depending on the transaction rate of the server.

Note The time criterion must be met for the server to be marked as dead.

Command Default

If this command is not used, the number of seconds ranges from 10 to 60 seconds, depending on the transaction rate of the server.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Note If you configure the **radius-server dead-criteria time** command before the **radius-server deadtime** command, the **radius-server dead-criteria time** command may not be enforced.

If a packet has not been received since the router booted and there is a timeout, the time criterion is treated as though it were met.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to establish the time for the dead-criteria conditions for a RADIUS server to be marked as dead for the **radius-server dead-criteria time** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server dead-criteria time 5
```

radius-server dead-criteria tries

To specify the number of consecutive timeouts that must occur on the router before the RADIUS server is marked as dead, use the **radius-server dead-criteria tries** command in the XR Config mode. To disable the criteria that were set, use the **no** form of this command.

radius-server dead-criteria *tries*
no radius-server dead-criteria *tries*

Syntax Description

tries Number of timeouts from 1 to 100. If the *tries* argument is not configured, the number of consecutive timeouts ranges from 10 to 100, depending on the transaction rate of the server and the number of configured retransmissions.

Note The tries criterion must be met for the server to be marked as dead.

Command Default

If this command is not used, the number of consecutive timeouts ranges from 10 to 100, depending on the transaction rate of the server and the number of configured retransmissions.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

If the server performs both authentication and accounting, both types of packet are included in the number. Improperly constructed packets are counted as though they were timeouts. All transmissions, including the initial transmit and all retransmits, are counted.

Note If you configure the **radius-server dead-criteria tries** command before the **radius-server deadtime** command, the **radius-server dead-criteria tries** command may not be enforced.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to establish the number of tries for the dead-criteria conditions for a RADIUS server to be marked as dead for the **radius-server dead-criteria tries** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server dead-criteria tries 4
```

radius-server deadtime (BNG)

To improve RADIUS response times when some servers are unavailable and cause the unavailable servers to be skipped immediately, use the **radius-server deadtime** command in the XR Config mode. To set deadtime to 0, use the **no** form of this command.

radius-server deadtime *value*
no radius-server deadtime *value*

Syntax Description

value	Length of time, in minutes, for which a RADIUS server is skipped over by transaction requests, up to a maximum of 1440 (24 hours). The range is from 1 to 1440. The default value is 0.

Command Default

Dead time is set to 0.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

A RADIUS server marked as dead is skipped by additional requests for the duration of minutes unless all other servers are marked dead and there is no rollover method.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example specifies five minutes of deadtime for RADIUS servers that fail to respond to authentication requests for the **radius-server deadtime** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server deadtime 5
```

radius-server key (BNG)

To set the authentication and encryption key for all RADIUS communications between the router and the RADIUS daemon, use the **radius-server key** command in the XR Config mode. To disable the key, use the **no** form of this command.

radius-server key {**0** *clear-text-key* | **7** *encrypted-keyclear-text-key*}
no radius-server key

Syntax Description

0 *clear-text-key*	Specifies an unencrypted (cleartext) shared key.
7 *encrypted-key*	Specifies a encrypted shared key.
clear-text-key	Specifies an unencrypted (cleartext) shared key.

Command Default

The authentication and encryption key is disabled.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The key entered must match the key used on the RADIUS server. All leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to set the cleartext key to “samplekey”:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server key 0 samplekey
```

This example shows how to set the encrypted shared key to “anykey”:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server key 7 anykey
```

radius-server retransmit (BNG)

To specify the number of times the Cisco IOS XR software retransmits a packet to a server before giving up, use the **radius-server retransmit** command in the XR Config mode. The **no** form of this command sets it to the default value of 3 .

radius-server retransmit {*retries* **disable**}
no radius-server retransmit {*retries* **disable**}

Syntax Description

retries	Maximum number of retransmission attempts. The range is from 1 to 100. Default is 3.
disable	Disables the radius-server transmit command.

Command Default

The RADIUS servers are retried three times, or until a response is received.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The RADIUS client tries all servers, allowing each one to time out before increasing the retransmit count.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to specify a retransmit counter value of five times:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server retransmit 5
```

radius-server timeout (BNG)

To set the interval for which a router waits for a server host to reply before timing out, use the **radius-server timeout** command in the XR Config mode. To restore the default, use the **no** form of this command.

radius-server timeout *seconds*
no radius-server timeout

Syntax Description

seconds	Number that specifies the timeout interval, in seconds. Range is from 1 to 1000.

Command Default

The default radius-server timeout value is 5 seconds.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **radius-server timeout** command to set the number of seconds a router waits for a server host to reply before timing out.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to change the interval timer to 10 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server timeout 10
```

radius source-interface (BNG)

To force RADIUS to use the IP address of a specified interface or subinterface for all outgoing RADIUS packets, use the **radius source-interface** command in the XR Config mode. To prevent only the specified interface from being the default and not from being used for all outgoing RADIUS packets, use the **no** form of this command.

radius source-interface *interface* [**vrf** *vrf_name*]
no radius source-interface *interface*

Syntax Description

interface-name	Name of the interface that RADIUS uses for all of its outgoing packets.
vrf *vrf-id*	Specifies the name of the assigned VRF.

Command Default

If a specific source interface is not configured, or the interface is down or does not have an IP address configured, the system selects an IP address.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **radius source-interface** command to set the IP address of the specified interface or subinterface for all outgoing RADIUS packets. This address is used as long as the interface or subinterface is in the up state. In this way, the RADIUS server can use one IP address entry for every network access client instead of maintaining a list of IP addresses.

The specified interface or subinterface must have an IP address associated with it. If the specified interface or subinterface does not have an IP address or is in the down state, then RADIUS reverts to the default. To avoid this, add an IP address to the interface or subinterface or bring the interface to the up state.

The **radius source-interface** command is especially useful in cases in which the router has many interfaces or subinterfaces and you want to ensure that all RADIUS packets from a particular router have the same IP address.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to make RADIUS use the IP address of subinterface s2 for all outgoing RADIUS packets:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius source-interface loopback 10 vrf vrf1
```

restrict-consecutive-characters

To restrict consecutive characters (that includes regular English alphabets, and English alphabets from QWERTY keyboard layout and numbers), for user passwords and secrets, use the **restrict-consecutive-characters** command in *aaa password-policy* configuration mode. To disable the feature, use the **no** form of the command.

restrict-consecutive-characters { **english-alphabet** | **qwerty-keyboard** } *num-of-chars* [**cyclic-wrap**]

Syntax Description

english-alphabet	Restricts consecutive English alphabets for user passwords and secrets. For example, "abcd", "wxyz", and so on.
qwerty-keyboard	Restricts consecutive English alphabets from QWERTY keyboard layout and numbers, for user passwords and secrets. For example, "qwer", "mnbv", "7890", and so on.
num-of-chars	Specifies the number of consecutive characters to be restricted for user passwords and secrets. Range is 2 to 26, for **english-alphabet**. Range is 2 to 10, for **qwerty-keyboard**.
cyclic-wrap	Restricts cyclic wrapping of the alphabet or the number for user passwords and secrets. For example, "yzab", "opqw", "9012", and so on.

Command Default

Disabled, by default.

Command Modes

aaa password-policy configuration mode

Command History

Release	Modification
Release 7.7.1	This command was introduced.

Usage Guidelines

All password policies are applicable only to locally configured users.

After creating the password policy, you must explicitly apply that policy to the user profiles so that the password policy take effect in the password and secret configuration.

For more details about the feature and configuration task, see the section *Enhanced Security for User Passwords and Secrets* in *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Among the NCS540 router variants, this command is applicable only for the following variants:

- N540-28Z4C-SYS-A/D
- N540X-16Z4G8Q2C-A/D
- N540-12Z20G-SYS-A/D

- N540X-12Z16G-SYS-A/D

Task ID

Task ID	Operation
aaa	read, write

This example shows how to configure a AAA password policy that restricts cyclic wrapping of four consecutive English alphabets and six consecutive characters from QWERTY keyboard.

```
Router(config)#aaa password-policy test-policy
Router(config-pp)#restrict-consecutive-characters english-alphabet 4 cyclic-wrap
Router(config-pp)#restrict-consecutive-characters qwerty-keyboard 6
```

This example shows how to apply the password policy to the user profile, *user1*:

```
Router(config)#username user1
Router(config-un)#policy test-policy
Router(config-un)#commit
```

Related Commands

Command	Description
aaa password-policy, on page 23	Defines the FIPS-compliant AAA password security policy.

secret

To configure an encrypted or clear-text password for the user, use the **secret** command in username configuration mode or line template configuration mode. To remove this configuration, use the **no** form of this command.

secret [{**0** [**enc-type** *enc-type-value*] | **5** | **8** | **9** | **10**}] *secret-login*
no secret

Syntax Description

0	(Optional) Specifies that an unencrypted (clear-text) password follows. The password will be encrypted for storage in the configuration using an MD5 encryption algorithm. Otherwise, the password is not encrypted.
5	Specifies that an encrypted MD5 password (secret) follows.
8	(Optional) Specifies that SHA256-encrypted password follows.
9	(Optional) Specifies that scrypt-encrypted password follows.
10	(Optional) Specifies that SHA512-encrypted password follows.
secret-login	Text string in alphanumeric characters that is stored as the MD5-encrypted password entered by the user in association with the user's login ID. Can be up to 253 characters in length. **Note** The characters entered must conform to MD5 encryption standards.
enc-type	(Optional) Configures the encryption type for a password entered in clear text.
enc-type-value	Specifies the encryption type to be used.

Command Default

No password is specified.

Command Modes

Username configuration

Line template configuration

Command History

Release	Modification
Release 6.0	This command was introduced.
Release 7.0.1	Added the support for Type 8 (SHA256), Type 9 (scrypt) and Type 10 (SHA512) encryption for **secret** configuration.
Release 7.0.1	Added the support for **enc-type** option under **secret 0** to specify the type of encryption for password entered in clear-text format.

Usage Guidelines

From Release 7.0.1 and later, Type 10 encryption is applied as the default encryption type for the **secret** on Cisco IOS XR 64-bit operating systems. Prior to this, Type 5 (MD5) was the default one.

Prior to Release 7.0.1, Cisco IOS XR software allows you to configure only Message Digest 5 (MD5) encryption for username logins and passwords. MD5 encryption is a one-way hash function that makes reversal of an encrypted password impossible, providing strong encryption protection. Using MD5 encryption, you cannot retrieve clear-text passwords. Therefore, MD5 encrypted passwords cannot be used with protocols that require the clear-text password to be retrievable, such as Challenge Handshake Authentication Protocol (CHAP).

Prior to Release 7.0.1, you can specify only one of two types of secure secret IDs: encrypted (5) or clear text (0). If you do not select either 0 or 5, the clear-text password you enter is not encrypted.

When an XR EXEC mode process is started on a line that has password protection, the process prompts for the secret. If the user enters the correct secret, the process issues the prompt. The user can try entering the secret thrice before the terminal returns to the idle state.

Secrets are one-way encrypted and should be used for login activities that do not require a decryptable secret.

To verify that MD5 password encryption has been enabled, use the **show running-config** command. The "username name secret 5" line in the command output indicates the same.

Note The **show running-config** command does not display the login password in clear text when the **0** option is used to specify an unencrypted password. See the "Examples" section.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to establish the clear-text secret "lab" for the user *user2*:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# username user2
RP/0/RP0/CPU0:router(config-un)# secret 0 lab
RP/0/RP0/CPU0:router(config-un)# commit
RP/0/RP0/CPU0:router(config-un)# show running-config
Building configuration...
username user2
 secret 5 $1$DTmd$q7C6fhzje7Cc7Xzmu2Frx1
!
end
```

The following examples show how to configure a Type 10 (SHA512) password for the user, *user10*. You can also see the examples and usage of the username, on page 108 command.

You can specify Type as '10' under the **secret** keyword, to explicitly configure Type 10 password.

```
Router#configure
Router(config)#username user10 secret 10
$6$9UvJidvsTEqgkAPU$3CL1Ei/F.E4v/Hi.UaqLwX8UsSEr9ApG6c5pzhMJmZtgW4jObAQ7meAwyhu5VM/aRFJqe/jxZG17h6xPrvJWf1
Router(config-un)#commit
```

You can also use the **enc-type** keyword under the **secret 0** option, to specify Type 10 as the encryption for a password entered in clear text.

```
Router#configure
Router(config)#username user10 secret 0 enc-type 10 testpassword
Router(config-un)#commit
```

server (RADIUS)

To associate a particular RADIUS server with a defined server group, use the **server** command in RADIUS server-group configuration mode. To remove the associated server from the server group, use the **no** form of this command.

server *ip-address* [**auth-port** *port-number*] [**acct-port** *port-number*]
no server *ip-address* [**auth-port** *port-number*] [**acct-port** *port-number*]

Syntax Description

ip-address	IP address of the RADIUS server host.
auth-port *port-number*	(Optional) Specifies the User Datagram Protocol (UDP) destination port for authentication requests. The *port-number* argument specifies the port number for authentication requests. The host is not used for authentication if this value is set to 0. Default is 1645.
acct-port *port-number*	(Optional) Specifies the UDP destination port for accounting requests. The *port-number* argument specifies the port number for accounting requests. The host is not used for accounting services if this value is set to 0. Default is 1646.

Command Default

If no port attributes are defined, the defaults are as follows:

- Authentication port: 1645
- Accounting port: 1646

Command Modes

RADIUS server-group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **server** command to associate a particular RADIUS server with a defined server group.

There are two different ways in which you can identify a server, depending on the way you want to offer AAA services. You can identify the server simply by using its IP address, or you can identify multiple host instances or entries using the optional **auth-port** and **acct-port** keywords.

When you use the optional keywords, the network access server identifies RADIUS security servers and host instances associated with a group server based on their IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS host entries providing a specific AAA service. If two different host entries on the same RADIUS server are configured for the same service, for example, accounting, the second host entry configured acts as an automatic switchover backup to the first one. Using this example, if the first host entry fails to provide accounting services, the network access server tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order they are configured.)

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to use two different host entries on the same RADIUS server that are configured for the same services—authentication and accounting. The second host entry configured acts as switchover backup to the first one.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa group server radius group1
RP/0/RP0/CPU0:router(config-sg-radius)# server 10.1.1.1 auth-port 1645 acct-port 1646
RP/0/RP0/CPU0:router(config-sg-radius)# server 10.2.2.2 auth-port 2000 acct-port 2001
```

server (TACACS+)

To associate a particular TACACS+ server with a defined server group, use the **server** command in TACACS+ server-group configuration mode. To remove the associated server from the server group, use the **no** form of this command.

server {*hostnameip-address*}
no server {*hostnameip-address*}

Syntax Description

hostname	Character string used to name the server host.
ip-address	IP address of the server host.

Command Default

None

Command Modes

TACACS+ server-group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **server** command to associate a particular TACACS+ server with a defined server group. The server need not be accessible during configuration. Later, you can reference the configured server group from the method lists used to configure authentication, authorization, and accounting (AAA).

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to associate the TACACS+ server with the IP address 192.168.60.15 with the server group tac1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa group server tacacs+ tac1
RP/0/RP0/CPU0:router(config-sg-tacacs+)# server 192.168.60.15
```

server-private (RADIUS)

To configure the IP address of the private RADIUS server for the group server, use the **server-private** command in RADIUS server-group configuration mode. To remove the associated private server from the AAA group server, use the **no** form of this command.

server-private *ip-address* [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
no server-private *ip-address* [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]

Syntax Description

ip-address	IP address of the RADIUS server host.
auth-port *port-number*	(Optional) Specifies the User Datagram Protocol (UDP) destination port for authentication requests. The *port-number* argument specifies the port number for authentication requests. The host is not used for authentication if this value is set to 0. The default value is 1645.
acct-port *port-number*	(Optional) Specifies the UDP destination port for accounting requests. The *port-number* argument specifies the port number for accounting requests. The host is not used for accounting services if this value is set to 0. The default value is 1646.
timeout *seconds*	(Optional) Specifies the number of seconds the router waits for the RADIUS server to reply before retransmitting. The setting overrides the global value of the **radius-server timeout** command. If no timeout is specified, the global value is used. The *seconds* argument specifies the timeout value in seconds. The range is from 1 to 1000. If no timeout is specified, the global value is used.
retransmit *retries*	(Optional) Specifies the number of times a RADIUS request is resent to a server if the server is not responding or is responding slowly. The setting overrides the global setting of the **radius-server transmit** command. The *retries* argument specifies the retransmit value. The range is from 1 to 100. If no retransmit value is specified, the global value is used.
key *string*	(Optional) Specifies the authentication and encryption key that is used between the router and the RADIUS daemon running on the RADIUS server. This key overrides the global setting of the **radius-server key** command. If no key string is specified, the global value is used.

Command Default

If no port attributes are defined, the defaults are as follows:

- Authentication port: 1645
- Accounting port: 1646

Command Modes

RADIUS server-group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **server-private** command to associate a particular private server with a defined server group. Possible overlapping of IP addresses between VRF instances are permitted. Private servers (servers with private addresses) can be defined within the server group and remain hidden from other groups, while the servers in the global pool (for example, default radius server group) can still be referred to by IP addresses and port numbers. Thus, the list of servers in server groups includes references to the hosts in the configuration and the definitions of private servers.

Both the **auth-port** and **acct-port** keywords enter RADIUS server-group private configuration mode.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to define the group1 RADIUS group server, to associate private servers with it, and to enter RADIUS server-group private configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa group server radius group1
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 timeout 5
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 retransmit 3
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 key coke
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 auth-port 300
RP/0/RP0/CPU0:router(config-sg-radius-private)# exit
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.2.2.2 timeout 5
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.2.2.2 retransmit 3
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.2.2.2 key coke
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.2.2.2 auth-port 300
RP/0/RP0/CPU0:router(config-sg-radius-private)#
```

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa group server radius group1
RP/0/RP0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 auth-port 300
RP/0/RP0/CPU0:router(config-sg-radius-private)# exit
(config-sg-radius)# server-private 10.2.2.2 auth-port 300
RP/0/RP0/CPU0:router(config-sg-radius-private)#
```

server-private (TACACS+)

To configure the IP address of the private TACACS+ server for the group server, use the **server-private** command in TACACS+ server-group configuration mode. To remove the associated private server from the AAA group server, use the **no** form of this command.

server-private {*hostnameip-address*} [**holddown-time** *time*][**port** *port-number*] [**timeout** *seconds*] [**key** *string*]
no server-private {*hostnameip-address*}

Syntax Description

hostname	Character string used to name the server host.
ip-address	IP address of the TACACS+ server host. Both IPv4 and IPv6 addresses are supported.
holddown-time *time*	Specifies a duration, in seconds, for which an unresponsive TACACS+ server is to be marked as DOWN. The range is from 0 to 1200. Zero indicates that the hold-down timer feature is disabled.
port *port-number*	(Optional) Specifies a server port number. This option overrides the default, which is port 49. Valid port numbers range from 1 to 65535.
timeout *seconds*	(Optional) Specifies, in seconds, a timeout value that sets the length of time the authentication, authorization, and accounting (AAA) server waits to receive a response from the TACACS+ server. This option overrides the global timeout value set with the **tacacs-server timeout** command for only this server. The range is from 1 to 1000. The default is 5.
key *string*	(Optional) Specifies the authentication and encryption key that is used between the router and the TACACS+ daemon running on the TACACS+ server. This key overrides the global setting of the **tacacs-server key** command. If no key string is specified, the global value is used.

Command Default

The *port-name* argument, if not specified, defaults to the standard port 49.

The *seconds* argument, if not specified, defaults to 5 seconds.

Command Modes

TACACS+ server-group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.
Release 7.4.1	This command was modified to include **holddown-time** option.

Usage Guidelines

Use the **server-private** command to associate a particular private server with a defined server group. Possible overlapping of IP addresses between VRF instances are permitted. Private servers (servers with private addresses) can be defined within the server group and remain hidden from other groups, while the servers in the global pool (for example, default tacacs+ server group) can still be referred by IP addresses and port

numbers. Therefore, the list of servers in server groups includes references to the hosts in the global configuration and the definitions of private servers.

For details on TACACS+ hold-down timer, see the **holddown-time** command.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to define the myserver TACACS+ group server, to associate private servers with it, and to enter TACACS+ server-group private configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa group server tacacs+ myserver
RP/0/RP0/CPU0:router(config-sg-tacacs+)# server-private 10.1.1.1 timeout 5
RP/0/RP0/CPU0:router(config-sg-tacacs+)# server-private 10.1.1.1 key a_secret
RP/0/RP0/CPU0:router(config-sg-tacacs+)# server-private 10.1.1.1 port 51
RP/0/RP0/CPU0:router(config-sg-tacacs-private)# exit
RP/0/RP0/CPU0:router(config-sg-tacacs+)# server-private 10.2.2.2 timeout 5
RP/0/RP0/CPU0:router(config-sg-tacacs+)# server-private 10.2.2.2 key coke
RP/0/RP0/CPU0:router(config-sg-tacacs+)# server-private 10.2.2.2 port 300
RP/0/RP0/CPU0:router(config-sg-tacacs-private)#
```

show aaa (XR-VM)

To display information about an Internet Key Exchange (IKE) Security Protocol group, user group, local user, login traces, or task group; to list all task IDs associated with all IKE groups, user groups, local users, or task groups in the system; or to list all task IDs for a specified IKE group, user group, local user, or task group, use the **show aaa** command in the XR EXEC mode.

show aaa {**ikegroup** *ikegroup-name* | **login sync** | **usergroup** [*usergroup-name*] | **trace** | **userdb** [*username*] | **task** | **taskgroup** }

Syntax Description

ikegroup	Displays details for local IKE groups.
ikegroup-name	(Optional) IKE group whose details are to be displayed.
login	Displays data for login subsystem.
sync	Syncs data with the subsystem.
usergroup	Displays details for all user groups.
usergroup-name	(Optional) Usergroup name.
trace	Displays trace data for AAA subsystem.
userdb	Displays details for all local users and the usergroups to which each user belongs.
username	(Optional) User whose details are to be displayed.
task	Show task information.
taskgroup	Displays details for all task groups. **Note** For taskgroup keywords, see optional usergroup name keyword list.

Command Default

Details for all user groups, or all local users, or all task groups are listed if no argument is entered.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show aaa** command to list details for all IKE groups, user groups, local users, AAA task IDs, or task groups in the system. Use the optional *ikegroup-name*, *usergroup-name*, *username* argument to display the details for a specified IKE group, user group, user, or task group, respectively.

Task ID

Task ID	Operations
aaa	read

Examples

The following sample output is from the **show aaa** command, using the **ikegroup** keyword:

```
RP/0/RP0/CPU0:router# show aaa ikegroup

IKE Group ike-group
        Max-Users = 50
IKE Group ikeuser
        Group-Key = test-password
        Default Domain = cisco.com
IKE Group ike-user
```

The following sample output is from the **show aaa** command, using the **usergroup** command:

```
RP/0/RP0/CPU0:router# show aaa usergroup operator

User group 'operator'
  Inherits from task group 'operator'
User group 'operator' has the following combined set
of task IDs (including all inherited groups):
Task:       basic-services  : READ    WRITE    EXECUTE DEBUG
Task:                 cdp  : READ
Task:                diag  : READ
Task:          ext-access  : READ             EXECUTE
Task:             logging  : READ
```

The following sample output is from the **show aaa** command, using the **taskgroup** keyword for a task group named netadmin:

```
RP/0/RP0/CPU0:router# show aaa taskgroup netadmin

Task group 'netadmin'

Task group 'netadmin' has the following combined set
  of task IDs (including all inherited groups):

Task:                 aaa  : READ
Task:                 acl  : READ    WRITE    EXECUTE    DEBUG
Task:               admin  : READ
Task:                ancp  : READ    WRITE    EXECUTE    DEBUG
Task:                 atm  : READ    WRITE    EXECUTE    DEBUG
Task:       basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:                bcdl  : READ
Task:                 bfd  : READ    WRITE    EXECUTE    DEBUG
Task:                 bgp  : READ    WRITE    EXECUTE    DEBUG
Task:                boot  : READ    WRITE    EXECUTE    DEBUG
Task:              bundle  : READ    WRITE    EXECUTE    DEBUG
Task:                 cdp  : READ    WRITE    EXECUTE    DEBUG
Task:                 cef  : READ    WRITE    EXECUTE    DEBUG
Task:                 cgn  : READ    WRITE    EXECUTE    DEBUG
Task:          config-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:      config-services  : READ    WRITE    EXECUTE    DEBUG
Task:              crypto  : READ    WRITE    EXECUTE    DEBUG
Task:                diag  : READ    WRITE    EXECUTE    DEBUG
Task:             drivers  : READ
Task:                dwdm  : READ    WRITE    EXECUTE    DEBUG
Task:                 eem  : READ    WRITE    EXECUTE    DEBUG
Task:    ethernet-services  : READ
Task:          ext-access  : READ    WRITE    EXECUTE    DEBUG
Task:              fabric  : READ    WRITE    EXECUTE    DEBUG
Task:           fault-mgr  : READ    WRITE    EXECUTE    DEBUG
Task:          filesystem  : READ    WRITE    EXECUTE    DEBUG
```

```
Task:              firewall  : READ    WRITE    EXECUTE    DEBUG
Task:                    fr  : READ    WRITE    EXECUTE    DEBUG
Task:                  hdlc  : READ    WRITE    EXECUTE    DEBUG
Task:        host-services  : READ    WRITE    EXECUTE    DEBUG
Task:                  hsrp  : READ    WRITE    EXECUTE    DEBUG
Task:             interface  : READ    WRITE    EXECUTE    DEBUG
Task:             inventory  : READ
Task:           ip-services  : READ    WRITE    EXECUTE    DEBUG
Task:                  ipv4  : READ    WRITE    EXECUTE    DEBUG
Task:                  ipv6  : READ    WRITE    EXECUTE    DEBUG
Task:                  isis  : READ    WRITE    EXECUTE    DEBUG
Task:                 l2vpn  : READ    WRITE    EXECUTE    DEBUG
Task:                    li  : READ    WRITE    EXECUTE    DEBUG
Task:               logging  : READ    WRITE    EXECUTE    DEBUG
Task:                  lpts  : READ    WRITE    EXECUTE    DEBUG
Task:               monitor  : READ
Task:              mpls-ldp  : READ    WRITE    EXECUTE    DEBUG
Task:           mpls-static  : READ    WRITE    EXECUTE    DEBUG
Task:               mpls-te  : READ    WRITE    EXECUTE    DEBUG
Task:             multicast  : READ    WRITE    EXECUTE    DEBUG
Task:               netflow  : READ    WRITE    EXECUTE    DEBUG
Task:               network  : READ    WRITE    EXECUTE    DEBUG
Task:                  ospf  : READ    WRITE    EXECUTE    DEBUG
Task:                  ouni  : READ    WRITE    EXECUTE    DEBUG
Task:              pkg-mgmt  : READ

Task:                   ppp  : READ    WRITE    EXECUTE    DEBUG
Task:                   qos  : READ    WRITE    EXECUTE    DEBUG
Task:                   rib  : READ    WRITE    EXECUTE    DEBUG
Task:                   rip  : READ    WRITE    EXECUTE    DEBUG
Task:               root-lr  : READ                                (reserved)
Task:             route-map  : READ    WRITE    EXECUTE    DEBUG
Task:          route-policy  : READ    WRITE    EXECUTE    DEBUG
Task:                   sbc  : READ    WRITE    EXECUTE    DEBUG
Task:                  snmp  : READ    WRITE    EXECUTE    DEBUG
Task:             sonet-sdh  : READ    WRITE    EXECUTE    DEBUG
Task:                static  : READ    WRITE    EXECUTE    DEBUG
Task:                sysmgr  : READ
Task:                system  : READ    WRITE    EXECUTE    DEBUG
Task:             transport  : READ    WRITE    EXECUTE    DEBUG
Task:           tty-access  : READ    WRITE    EXECUTE    DEBUG
Task:                tunnel  : READ    WRITE    EXECUTE    DEBUG
Task:             universal  : READ                                (reserved)
Task:                  vlan  : READ    WRITE    EXECUTE    DEBUG
Task:                  vrrp  : READ    WRITE    EXECUTE    DEBUG
```

The following sample output is from the **show aaa** command, using the **taskgroup** keyword for an operator. The task group operator has the following combined set of task IDs, which includes all inherited groups:

```
Task:       basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:                  cdp  : READ
Task:                 diag  : READ
Task:           ext-access  : READ             EXECUTE
Task:              logging  : READ
```

The following sample output is from the **show aaa** command, using the **taskgroup** keyword for a root system. The task-group root system has the following combined set of task IDs, which includes all inherited groups:

```
Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  aaa acl  : READ    WRITE    EXECUTE    DEBUG
```

```
Task:                  acl admin  : READ    WRITE    EXECUTE    DEBUG
Task:                  admin atm  : READ    WRITE    EXECUTE    DEBUG
Task:                  atm basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:       basic-services           bcdl  : READ    WRITE    EXECUTE    DEBUG
Task:                bcdl  bfd  : READ    WRITE    EXECUTE    DEBUG
Task:                 bfd bgp  : READ    WRITE    EXECUTE    DEBUG
Task:                 bgp boot  : READ    WRITE    EXECUTE    DEBUG
Task:                boot bundle  : READ    WRITE    EXECUTE    DEBUG
Task:              bundle    cdp  : READ    WRITE    EXECUTE    DEBUG
Task:                 cdp cef  : READ    WRITE    EXECUTE    DEBUG
Task:                 cef config-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:          config-mgmt services  : READ    WRITE    EXECUTE    DEBUG
Task:      config-services           crypto  : READ    WRITE    EXECUTE    DEBUG
Task:               crypto   diag  : READ    WRITE    EXECUTE    DEBUG
Task:                 diag  drivers  : READ    WRITE    EXECUTE    DEBUG
Task:              drivers ext-access  : READ    WRITE    EXECUTE    DEBUG
Task:          ext-access     fabric  : READ    WRITE    EXECUTE    DEBUG
Task:              fabric  fault-mgr  : READ    WRITE    EXECUTE    DEBUG
Task:           fault-mgr filesystem  : READ    WRITE    EXECUTE    DEBUG
Task:          filesystem         fr  : READ    WRITE    EXECUTE    DEBUG
Task:                  fr hdlc  : READ    WRITE    EXECUTE    DEBUG
Task:                hdlc  host-services  : READ    WRITE    EXECUTE    DEBUG
Task:        host-services           hsrp  : READ    WRITE    EXECUTE    DEBUG
Task:                hsrp  interface  : READ    WRITE    EXECUTE    DEBUG
Task:           interface inventory  : READ    WRITE    EXECUTE    DEBUG
Task:           inventory ip-services  : READ    WRITE    EXECUTE    DEBUG
Task:          ip-services         ipv4  : READ    WRITE    EXECUTE    DEBUG
Task:                ipv4 ipv6  : READ    WRITE    EXECUTE    DEBUG
Task:                ipv6 isis  : READ    WRITE    EXECUTE    DEBUG
Task:                isis  logging  : READ    WRITE    EXECUTE    DEBUG
Task:             logging     lpts  : READ    WRITE    EXECUTE    DEBUG
Task:                lpts  monitor  : READ    WRITE    EXECUTE    DEBUG
Task:             monitor mpls-ldp  : READ    WRITE    EXECUTE    DEBUG
Task:             mpls-ldp static  : READ    WRITE    EXECUTE    DEBUG
Task:             mpls-static te  : READ    WRITE    EXECUTE    DEBUG
Task:             mpls-te multicast  : READ    WRITE    EXECUTE    DEBUG
Task:             multicast netflow  : READ    WRITE    EXECUTE    DEBUG
Task:             netflow network  : READ    WRITE    EXECUTE    DEBUG
Task:             network     ospf  : READ    WRITE    EXECUTE    DEBUG
Task:                ospf ouni  : READ    WRITE    EXECUTE    DEBUG
Task:                ouni pkg-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:            pkg  mgmt   : READ    WRITE    EXECUTE    DEBUG
Task:                 ppp  : READ    WRITE    EXECUTE    DEBUG
Task:                 qos  : READ    WRITE    EXECUTE    DEBUG
Task:                 rib  : READ    WRITE    EXECUTE    DEBUG
Task:                 rip  : READ    WRITE    EXECUTE    DEBUG
Task:             root-lr  : READ    WRITE    EXECUTE    DEBUG
Task:          root-system  : READ    WRITE    EXECUTE    DEBUG
Task:            route-map  : READ    WRITE    EXECUTE    DEBUG
Task:         route-policy  : READ    WRITE    EXECUTE    DEBUG
Task:                 snmp  : READ    WRITE    EXECUTE    DEBUG
Task:           sonet-sdh  : READ    WRITE    EXECUTE    DEBUG
Task:              static  : READ    WRITE    EXECUTE    DEBUG
Task:              sysmgr  : READ    WRITE    EXECUTE    DEBUG
Task:              system  : READ    WRITE    EXECUTE    DEBUG
Task:           transport  : READ    WRITE    EXECUTE    DEBUG
Task:           tty-access  : READ    WRITE    EXECUTE    DEBUG
Task:              tunnel  : READ    WRITE    EXECUTE    DEBUG
Task:            universal  : READ    WRITE    EXECUTE    DEBUG
Task:                 vlan  : READ    WRITE    EXECUTE    DEBUG
Task:                 vrrp  : READ    WRITE    EXECUTE    DEBUG
```

The following sample output is from the **show aaa** command, using the **task supported** keywords. Task IDs are displayed in alphabetic order.

```
RP/0/RP0/CPU0:router# show aaa task supported

aaa
acl
admin
atm
basic-services
bcdl
bfd
bgp
boot
bundle
cdp
cef
cisco-support
config-mgmt
config-services
crypto
diag
disallowed
drivers
ext-access
fabric
fault-mgr
filesystem
firewall
fr
hdlc
host-services
hsrp
interface
inventory
ip-services
ipv4
ipv6
isis
logging
lpts
monitor
mpls-ldp
mpls-static
mpls-te
multicast
netflow
network
ospf
ouni
pkg-mgmt

ppp
qos
rib
rip
User group root-systemlrlr
root-system
route-map
route-policy
sbc
snmp
sonet-sdh
static
sysmgr
system
```

```
transport
tty-access
tunnel
universal
vlan
vrrp
```

show aaa accounting

To display command history with the date and time for AAA sub-system, use the **show aaa accounting** command in the System Admin EXEC mode. You must have a group aaa-r or root-system on System Admin VM.

show aaa accounting

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

System Admin EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
aaa	read

This is the sample output of the **show aaa accounting** command:

```
sysadmin-vm:0_RP0#show aaa accounting
Mon Nov  3  13:37:21.573 UTC

Detail audit log information
  Time                      Username          Session-ID     Node-Information     Command
-------------------------------------------------------------------------------------------------
2014-11-03.13:14:27 UTC      root                  17         System        logged in from
 the CLI with aaa disabled
..
…
2014-11-03.13:37:01 UTC      cisco                 57          0/RP0         assigned to
groups: root-system
2014-11-03.13:37:03 UTC      cisco                 57          0/RP0         CLI 'config
terminal'
2014-11-03.13:37:03 UTC      cisco                 57          0/RP0         CLI done
2014-11-03.13:37:09 UTC      cisco                 57          0/RP0         CLI 'aaa
authentication users user temp'
2014-11-03.13:37:09 UTC      cisco                 57          0/RP0         CLI done
2014-11-03.13:37:11 UTC      cisco                 57          0/RP0         CLI 'password
 ****
2014-11-03.13:37:11 UTC      cisco                 57          0/RP0         CLI done
2014-11-03.13:37:12 UTC      cisco                 57          0/RP0         CLI 'commit'
2014-11-03.13:37:14 UTC      cisco                 57          0/RP0         CLI done
2014-11-03.13:37:16 UTC      cisco                 57          0/RP0         CLI 'exit'
2014-11-03.13:37:16 UTC      cisco                 57          0/RP0         CLI done
2014-11-03.13:37:18 UTC      cisco                 57          0/RP0         CLI 'exit'
2014-11-03.13:37:18 UTC      cisco                 57          0/RP0         CLI done
```

```
2014-11-03.13:37:21 UTC      cisco                  57          0/RP0          CLI 'show aaa
 accounting'
```

show aaa password-policy

To display the details of AAA password policy configured in a system, use the **show aaa password-policy** command in XR EXEC mode.

show aaa password-policy [*policy-name*]

Syntax Description

policy-name	Specifies the name of password policy.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.2.1	This command was introduced.

Usage Guidelines

If the option *policy-name* is not specified, the command output displays the details of all password policies configured in the system.

Refer **aaa password-policy** command details of each field in this command output.

Task ID

Task ID	Operation
aaa	read

This is a sample out of **show aaa password-policy** command:

```
RP/0/RP0/CPU0:router#show aaa password-policy test-policy

Fri Feb  3 16:50:58.086 EDT
Password Policy Name : test-policy
  Number of Users : 1
  Minimum Length : 2
  Maximum Length : 253
  Special Character Len : 0
  Uppercase Character Len : 0
  Lowercase Character Len : 1
  Numeric Character Len : 0
  Policy Life Time :
    seconds : 0
    minutes : 0
    hours : 0
    days : 0
    months : 0
    years : 0
  Lockout Time :
    seconds : 0
    minutes : 0
    hours : 0
    days : 0
```

```
  months : 0
  years : 0
Character Change Len : 4
Maximum Failure Attempts : 0
```

Related Commands

Command	Description
aaa password-policy, on page 23	Defines the FIPS-compliant AAA password security policy.

show radius

To display information about the RADIUS servers that are configured in the system, use the **show radius** command in the XR EXEC mode.

show radius

Syntax Description

This command has no keywords or arguments.

Command Default

If no radius servers are configured, no output is displayed.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show radius** command to display statistics for each configured RADIUS server.

Task ID

Task ID	Operations
aaa	read

Examples

The following sample output is for the **show radius** command:

```
RP/0/RP0/CPU0:router# show radius

Global dead time: 0 minute(s)

Server: 10.1.1.1/1645/1646  is UP
  Timeout: 5 sec, Retransmit limit: 3
  Quarantined: No
  Authentication:
    0 requests, 0 pending, 0 retransmits
    0 accepts, 0 rejects, 0 challenges
    0 timeouts, 0 bad responses, 0 bad authenticators
    0 unknown types, 0 dropped, 0 ms latest rtt
  Accounting:
    0 requests, 0 pending, 0 retransmits
    0 responses, 0 timeouts, 0 bad responses
    0 bad authenticators, 0 unknown types, 0 dropped
    0 ms latest rtt

Server: 10.2.2.2/1645/1646  is UP
  Timeout: 10 sec, Retransmit limit: 3
  Authentication:
    0 requests, 0 pending, 0 retransmits
    0 accepts, 0 rejects, 0 challenges
    0 timeouts, 0 bad responses, 0 bad authenticators
    0 unknown types, 0 dropped, 0 ms latest rtt
  Accounting:
    0 requests, 0 pending, 0 retransmits
```

```
    0 responses, 0 timeouts, 0 bad responses
    0 bad authenticators, 0 unknown types, 0 dropped
    0 ms latest rtt
```

This table describes the significant fields shown in the display.

Table 2: show radius Field Descriptions

Field	Description
Server	Server IP address/UDP destination port for authentication requests/UDP destination port for accounting requests.
Timeout	Number of seconds the router waits for a server host to reply before timing out.
Retransmit limit	Number of times the Cisco IOS XR software searches the list of RADIUS server hosts before giving up.

show radius accounting

To obtain information and detailed statistics for the RADIUS accounting server and port, use the **show radius accounting** command in the XR EXEC mode

show radius accounting

Syntax Description

This command has no keywords or arguments.

Command Default

If no RADIUS servers are configured on the router, the output is empty. If the default values are for the counter (for example, request and pending), the values are all zero because the RADIUS server was just defined and not used yet.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
aaa	read

Examples

The following sample output is displayed on a per-server basis for the **show radius accounting** command:

```
RP/0/RP0/CPU0:router# show radius accounting

Server: 12.26.25.61, port: 1813
0 requests, 0 pending, 0 retransmits
0 responses, 0 timeouts, 0 bad responses
0 bad authenticators, 0 unknown types, 0 dropped
0 ms latest rtt

Server: 12.26.49.12, port: 1813
0 requests, 0 pending, 0 retransmits
0 responses, 0 timeouts, 0 bad responses
0 bad authenticators, 0 unknown types, 0 dropped
0 ms latest rtt

Server: 12.38.28.18, port: 29199
0 requests, 0 pending, 0 retransmits
0 responses, 0 timeouts, 0 bad responses
0 bad authenticators, 0 unknown types, 0 dropped
0 ms latest rtt
```

This table describes the significant fields shown in the display.

Table 3: show radius accounting Field Descriptions

Field	Description
Server	Server IP address/UDP destination port for authentication requests; UDP destination port for accounting requests.

show radius authentication

To obtain information and detailed statistics for the RADIUS authentication server and port, use the **show radius authentication** command in the XR EXEC mode.

show radius authentication

Syntax Description

This command has no keywords or arguments.

Command Default

If no RADIUS servers are configured on the router, the output is empty. If the default values are for the counter (for example, request and pending), the values are all zero because the RADIUS server was just defined and not used yet.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
aaa	read

Examples

The following sample output is for the **show radius authentication** command:

```
RP/0/RP0/CPU0:router# show radius authentication

Server: 12.26.25.61, port: 1812
0 requests, 0 pending, 0 retransmits
0 accepts, 0 rejects, 0 challenges
0 timeouts, 0 bad responses, 0 bad authenticators
0 unknown types, 0 dropped, 0 ms latest rtt

Server: 12.26.49.12, port: 1812
0 requests, 0 pending, 0 retransmits
0 accepts, 0 rejects, 0 challenges
0 timeouts, 0 bad responses, 0 bad authenticators
0 unknown types, 0 dropped, 0 ms latest rtt

Server: 12.38.28.18, port: 21099
0 requests, 0 pending, 0 retransmits
0 accepts, 0 rejects, 0 challenges
0 timeouts, 0 bad responses, 0 bad authenticators
0 unknown types, 0 dropped, 0 ms latest rtt
```

This table describes the significant fields shown in the display.

Table 4: show radius authentication Field Descriptions

Field	Description
Server	Server IP address/UDP destination port for authentication requests; UDP destination port for accounting requests.

show radius dead-criteria

To obtain information about the dead server detection criteria, use the **show radius dead-criteria** command in the XR EXEC mode.

show radius dead-criteria host *ip-addr* [**auth-port** *auth-port*] [**acct-port** *acct-port*]

Syntax Description

host *ip-addr*	Specifies the name or IP address of the configured RADIUS server.
auth-port *auth-port*	(Optional) Specifies the authentication port for the RADIUS server. The default value is 1645.
acct-port *acct-port*	(Optional) Specifies the accounting port for the RADIUS server. The default value is 1646.

Command Default

The default values for time and tries are not fixed to a single value; therefore, they are calculated and fall within a range of 10 to 60 seconds for time and 10 to 100 for tries.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
aaa	read

Examples

The following sample output is for the **show radius dead-criteria** command:

```
RP/0/RP0/CPU0:router# show radius dead-criteria host 12.26.49.12 auth-port 11000 acct-port
 11001

Server: 12.26.49.12/11000/11001
Dead criteria time: 10 sec (computed) tries: 10 (computed)
```

This table describes the significant fields shown in the display.

Table 5: show radius dead-criteria Field Descriptions

Field	Description
Server	Server IP address/UDP destination port for authentication requests/UDP destination port for accounting requests.
Timeout	Number of seconds the router waits for a server host to reply before timing out.

Field	Description
Retransmits	Number of times Cisco IOS XR software searches the list of RADIUS server hosts before giving up.

show radius server-groups

To display information about the RADIUS server groups that are configured in the system, use the **show radius server-groups** command in the XR EXEC mode.

show radius server-groups [*group-name* [**detail**]]

Syntax Description

group-name	(Optional) Name of the server group. The properties are displayed.
detail	(Optional) Displays properties for all the server groups.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show radius server-groups** command to display information about each configured RADIUS server group, including the group name, numbers of servers in the group, and a list of servers in the named server group. A global list of all configured RADIUS servers, along with authentication and accounting port numbers, is also displayed.

Task ID

Task ID	Operations
aaa	read

Examples

The inherited global message is displayed if no group level deadtime is defined for this group; otherwise, the group level deadtime value is displayed and this message is omitted. The following sample output is for the **show radius server-groups** command:

```
RP/0/RP0/CPU0:router# show radius server-groups

Global list of servers
  Contains 2 server(s)
    Server 10.1.1.1/1645/1646
    Server 10.2.2.2/1645/1646

Server group 'radgrp1' has 2 server(s)
  Dead time: 0 minute(s) (inherited from global)
  Contains 2 server(s)
    Server 10.1.1.1/1645/1646
    Server 10.2.2.2/1645/1646

Server group 'radgrp-priv' has 1 server(s)
  Dead time: 0 minute(s) (inherited from global)
  Contains 1 server(s)
    Server 10.3.3.3/1645/1646 [private]
```

The following sample output shows the properties for all the server groups in group “radgrp1:”

```
RP/0/RP0/CPU0:router# show radius server-groups radgrp1 detail

Server group 'radgrp1' has 2 server(s)
    VRF default (id 0x60000000)
    Dead time: 0 minute(s) (inherited from global)
    Contains 2 server(s)
      Server 10.1.1.1/1645/1646
    Authentication:
      0 requests, 0 pending, 0 retransmits
      0 accepts, 0 rejects, 0 challenges
      0 timeouts, 0 bad responses, 0 bad authenticators
      0 unknown types, 0 dropped, 0 ms latest rtt
    Accounting:
      0 requests, 0 pending, 0 retransmits
      0 responses, 0 timeouts, 0 bad responses
      0 bad authenticators, 0 unknown types, 0 dropped
      0 ms latest rtt
      Server 10.2.2.2/1645/1646
    Authentication:
      0 requests, 0 pending, 0 retransmits
      0 accepts, 0 rejects, 0 challenges
      0 timeouts, 0 bad responses, 0 bad authenticators
      0 unknown types, 0 dropped, 0 ms latest rtt
    Accounting:
      0 requests, 0 pending, 0 retransmits
      0 responses, 0 timeouts, 0 bad responses
      0 bad authenticators, 0 unknown types, 0 dropped
      0 ms latest rtt
```

The following sample output shows the properties for all the server groups in detail in the group “raddgrp-priv:”

```
RP/0/RP0/CPU0:router# show radius server-groups radgrp-priv detail

Server group 'radgrp-priv' has 1 server(s)
    VRF default (id 0x60000000)
    Dead time: 0 minute(s) (inherited from global)
    Contains 1 server(s)
      Server 10.3.3.3/1645/1646 [private]
    Authentication:
      0 requests, 0 pending, 0 retransmits
      0 accepts, 0 rejects, 0 challenges
      0 timeouts, 0 bad responses, 0 bad authenticators
      0 unknown types, 0 dropped, 0 ms latest rtt
    Accounting:
      0 requests, 0 pending, 0 retransmits
      0 responses, 0 timeouts, 0 bad responses
      0 bad authenticators, 0 unknown types, 0 dropped
      0 ms latest rtt
```

This table describes the significant fields shown in the display.

Table 6: show radius server-groups Field Descriptions

Field	Description
Server	Server IP address/UDP destination port for authentication requests/UDP destination port for accounting requests.

show tacacs

To display information about the TACACS+ servers that are configured in the system, use the **show tacacs** command in the XR EXEC mode.

show tacacs

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show tacacs** command to display statistics for each configured TACACS+ server.

Task ID

Task ID	Operations
aaa	read

Examples

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

```
RP/0/RP0/CPU0:router# show tacacs

For IPv4 IP addresses:
Server:10.1.1.1/21212 opens=0 closes=0 aborts=0 errors=0
        packets in=0 packets out=0
        status=up single-connect=false

Server:10.2.2.2/21232 opens=0 closes=0 aborts=0 errors=0
        packets in=0 packets out=0
        status=up single-connect=false

For IPv6 IP addresses:
Server: 10.2.3.5/49  family = AF_INET opens=0 closes=0 aborts=0 errors=0
        packets in=0 packets out=0
        status=up single-connect=false
```

This table describes the significant fields shown in the display.

Table 7: show tacacs Field Descriptions

Field	Description
Server	Server IP address.
opens	Number of socket opens to the external server.

Field	Description
closes	Number of socket closes to the external server.
aborts	Number of tacacs requests that have been terminated midway.
errors	Number of error replies from the external server.
packets in	Number of TCP packets that have been received from the external server.
packets out	Number of TCP packets that have been sent to the external server.

show tacacs server-groups

To display information about the TACACS+ server groups that are configured in the system, use the **show tacacs server-groups** command in the XR EXEC mode.

show tacacs server-groups

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show tacacs server-groups** command to display information about each configured TACACS+ server group, including the group name, numbers of servers in the group, and a list of servers in the named server group. A global list of all configured TACACS+ servers is also displayed.

Task ID

Task ID	Operations
aaa	read

Examples

The following is sample output from the **show tacacs server-groups** command:

```
RP/0/RP0/CPU0:router# show tacacs server-groups

Global list of servers
  Server 192.168.25.61/23456
  Server 192.168.49.12/12345
  Server 192.168.49.12/9000
  Server 192.168.25.61/23432
  Server 10.5.5.5/23456
  Server 10.1.1.1/49
Server group 'tac100' has 1 servers
Server 192.168.49.12
```

This table describes the significant fields shown in the display.

Table 8: show tacacs server-groups Field Descriptions

Field	Description
Server	Server IP address.

show user

To display all user groups and task IDs associated with the currently logged-in user, use the **show user** command in the XR EXEC mode.

show user [{**all** | **authentication** | **group** | **tasks**}]

Syntax Description

all	(Optional) Displays all user groups and task IDs for the currently logged-in user.
authentication	(Optional) Displays authentication method parameters for the currently logged-in user.
group	(Optional) Displays the user groups associated with the currently logged-in user.
tasks	(Optional) Displays task IDs associated with the currently logged-in user. The **tasks** keyword indicates which task is reserved in the sample output.

Command Default

When the **show user** command is used without any option, it displays the ID of the user who is logged in currently.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show user** command to display all user groups and task IDs associated with the currently logged-in user.

Task ID

Task ID	Operations
none	—

Examples

The following sample output displays the authentication method parameters from the **show user** command:

```
RP/0/RP0/CPU0:router# show user authentication method

local
```

The following sample output displays the groups from the **show user** command:

```
RP/0/RP0/CPU0:router# show user group

root-system
```

The following sample output displays all the information for the groups and tasks from the **show user** command:

```
RP/0/RP0/CPU0:router# show user all
Username: lab
Groups: root-system
Authenticated using method local
User lab has the following Task ID(s):

Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  acl  : READ    WRITE    EXECUTE    DEBUG
Task:                admin  : READ    WRITE    EXECUTE    DEBUG
Task:                  atm  : READ    WRITE    EXECUTE    DEBUG
Task:       basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 bcdl  : READ    WRITE    EXECUTE    DEBUG
Task:                  bfd  : READ    WRITE    EXECUTE    DEBUG
Task:                  bgp  : READ    WRITE    EXECUTE    DEBUG
Task:                 boot  : READ    WRITE    EXECUTE    DEBUG
Task:               bundle  : READ    WRITE    EXECUTE    DEBUG
Task:                  cdp  : READ    WRITE    EXECUTE    DEBUG
Task:                  cef  : READ    WRITE    EXECUTE    DEBUG
Task:          config-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:      config-services  : READ    WRITE    EXECUTE    DEBUG
Task:               crypto  : READ    WRITE    EXECUTE    DEBUG
Task:                 diag  : READ    WRITE    EXECUTE    DEBUG
Task:              drivers  : READ    WRITE    EXECUTE    DEBUG
Task:           ext-access  : READ    WRITE    EXECUTE    DEBUG
Task:               fabric  : READ    WRITE    EXECUTE    DEBUG
Task:            fault-mgr  : READ    WRITE    EXECUTE    DEBUG
Task:           filesystem  : READ    WRITE    EXECUTE    DEBUG
Task:             firewall  : READ    WRITE    EXECUTE    DEBUG
Task:                   fr  : READ    WRITE    EXECUTE    DEBUG
Task:                 hdlc  : READ    WRITE    EXECUTE    DEBUG
Task:        host-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 hsrp  : READ    WRITE    EXECUTE    DEBUG
Task:            interface  : READ    WRITE    EXECUTE    DEBUG
Task:            inventory  : READ    WRITE    EXECUTE    DEBUG
Task:          ip-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 ipv4  : READ    WRITE    EXECUTE    DEBUG
Task:                 ipv6  : READ    WRITE    EXECUTE    DEBUG
Task:                 isis  : READ    WRITE    EXECUTE    DEBUG
Task:              logging  : READ    WRITE    EXECUTE    DEBUG
Task:                 lpts  : READ    WRITE    EXECUTE    DEBUG
Task:              monitor  : READ    WRITE    EXECUTE    DEBUG
Task:              mpls-ldp  : READ    WRITE    EXECUTE    DEBUG
Task:              mpls-static  : READ    WRITE    EXECUTE    DEBUG
Task:              mpls-te  : READ    WRITE    EXECUTE    DEBUG
Task:              multicast  : READ    WRITE    EXECUTE    DEBUG
Task:              netflow  : READ    WRITE    EXECUTE    DEBUG
Task:              network  : READ    WRITE    EXECUTE    DEBUG
Task:                 ospf  : READ    WRITE    EXECUTE    DEBUG
Task:                 ouni  : READ    WRITE    EXECUTE    DEBUG
Task:             pkg-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:                  ppp  : READ    WRITE    EXECUTE    DEBUG
Task:                  qos  : READ    WRITE    EXECUTE    DEBUG
Task:                  rib  : READ    WRITE    EXECUTE    DEBUG
Task:                  rip  : READ    WRITE    EXECUTE    DEBUG
Task:              root-lr  : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:           root-system  : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:            route-map  : READ    WRITE    EXECUTE    DEBUG
Task:         route-policy  : READ    WRITE    EXECUTE    DEBUG
Task:                  sbc  : READ    WRITE    EXECUTE    DEBUG
Task:                 snmp  : READ    WRITE    EXECUTE    DEBUG
Task:            sonet-sdh  : READ    WRITE    EXECUTE    DEBUG
Task:               static  : READ    WRITE    EXECUTE    DEBUG
```

```
Task:               sysmgr  : READ    WRITE    EXECUTE    DEBUG
Task:               system  : READ    WRITE    EXECUTE    DEBUG
Task:            transport  : READ    WRITE    EXECUTE    DEBUG
Task:          tty-access  : READ    WRITE    EXECUTE    DEBUG
Task:               tunnel  : READ    WRITE    EXECUTE    DEBUG
Task:            universal  : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:                 vlan  : READ    WRITE    EXECUTE    DEBUG
Task:                 vrrp  : READ    WRITE    EXECUTE    DEBUG
```

The following sample output displays the tasks and indicates which tasks are reserved from the **show user** command:

```
RP/0/RP0/CPU0:router# show user tasks

Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  aaa  : READ    WRITE    EXECUTE    DEBUG
Task:                  acl  : READ    WRITE    EXECUTE    DEBUG
Task:                admin  : READ    WRITE    EXECUTE    DEBUG
Task:                  atm  : READ    WRITE    EXECUTE    DEBUG
Task:       basic-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 bcdl  : READ    WRITE    EXECUTE    DEBUG
Task:                  bfd  : READ    WRITE    EXECUTE    DEBUG
Task:                  bgp  : READ    WRITE    EXECUTE    DEBUG
Task:                 boot  : READ    WRITE    EXECUTE    DEBUG
Task:               bundle  : READ    WRITE    EXECUTE    DEBUG
Task:                  cdp  : READ    WRITE    EXECUTE    DEBUG
Task:                  cef  : READ    WRITE    EXECUTE    DEBUG
Task:          config-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:      config-services  : READ    WRITE    EXECUTE    DEBUG
Task:               crypto  : READ    WRITE    EXECUTE    DEBUG
Task:                 diag  : READ    WRITE    EXECUTE    DEBUG
Task:              drivers  : READ    WRITE    EXECUTE    DEBUG
Task:           ext-access  : READ    WRITE    EXECUTE    DEBUG
Task:               fabric  : READ    WRITE    EXECUTE    DEBUG
Task:            fault-mgr  : READ    WRITE    EXECUTE    DEBUG
Task:           filesystem  : READ    WRITE    EXECUTE    DEBUG
Task:             firewall  : READ    WRITE    EXECUTE    DEBUG
Task:                   fr  : READ    WRITE    EXECUTE    DEBUG
Task:                 hdlc  : READ    WRITE    EXECUTE    DEBUG
Task:        host-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 hsrp  : READ    WRITE    EXECUTE    DEBUG
Task:            interface  : READ    WRITE    EXECUTE    DEBUG
Task:            inventory  : READ    WRITE    EXECUTE    DEBUG
Task:          ip-services  : READ    WRITE    EXECUTE    DEBUG
Task:                 ipv4  : READ    WRITE    EXECUTE    DEBUG
Task:                 ipv6  : READ    WRITE    EXECUTE    DEBUG
Task:                 isis  : READ    WRITE    EXECUTE    DEBUG
Task:              logging  : READ    WRITE    EXECUTE    DEBUG
Task:                 lpts  : READ    WRITE    EXECUTE    DEBUG
Task:              monitor  : READ    WRITE    EXECUTE    DEBUG
Task:              mpls-ldp  : READ    WRITE    EXECUTE    DEBUG
Task:              mpls-static  : READ    WRITE    EXECUTE    DEBUG
Task:              mpls-te  : READ    WRITE    EXECUTE    DEBUG
Task:              multicast  : READ    WRITE    EXECUTE    DEBUG
Task:              netflow  : READ    WRITE    EXECUTE    DEBUG
Task:              network  : READ    WRITE    EXECUTE    DEBUG
Task:                 ospf  : READ    WRITE    EXECUTE    DEBUG
Task:                 ouni  : READ    WRITE    EXECUTE    DEBUG
Task:              pkg-mgmt  : READ    WRITE    EXECUTE    DEBUG
Task:                  ppp  : READ    WRITE    EXECUTE    DEBUG
Task:                  qos  : READ    WRITE    EXECUTE    DEBUG
Task:                  rib  : READ    WRITE    EXECUTE    DEBUG
Task:                  rip  : READ    WRITE    EXECUTE    DEBUG
```

```
Task:              root-lr  : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:         root-system   : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:           route-map   : READ    WRITE    EXECUTE    DEBUG
Task:        route-policy   : READ    WRITE    EXECUTE    DEBUG
Task:                 sbc   : READ    WRITE    EXECUTE    DEBUG
Task:                snmp   : READ    WRITE    EXECUTE    DEBUG
Task:           sonet-sdh   : READ    WRITE    EXECUTE    DEBUG
Task:              static   : READ    WRITE    EXECUTE    DEBUG
Task:              sysmgr   : READ    WRITE    EXECUTE    DEBUG
Task:              system   : READ    WRITE    EXECUTE    DEBUG
Task:           transport   : READ    WRITE    EXECUTE    DEBUG
Task:         tty-access    : READ    WRITE    EXECUTE    DEBUG
Task:              tunnel   : READ    WRITE    EXECUTE    DEBUG
Task:           universal   : READ    WRITE    EXECUTE    DEBUG (reserved)
Task:                vlan   : READ    WRITE    EXECUTE    DEBUG
Task:                vrrp   : READ    WRITE    EXECUTE    DEBUG
```

show aaa user-group

To display user group information for AAA sub-system, use the **show aaa user-group** command in the System Admin EXEC mode. You must have a group aaa-r or root-system on System Admin VM.

show aaa user-group

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

System Admin EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
aaa	read

This is the sample output of the **show aaa user-group** command:

```
sysadmin-vm:0_RP0#show aaa user-group
Mon Nov  3  13:39:33.380 UTC

User group : root-system
sysadmin-vm:0_RP0#
```

show tech-support aaa

To collect AAA debug and trace files from System Admin VM, use the **show tech-support aaa** command in the System Admin EXEC mode.

show tech-support aaa

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

System Admin EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
aaa	read

This is the sample output of the **show tech-support aaa** command:

```
sysadmin-vm:0_RP0#show tech-support aaa
Mon Nov  3  13:39:33.380 UTC

Fri Oct  24 07:22:15.740 UTC ++ Show tech start time: 2014-Oct-24.072216.UTC ++
Waiting for gathering to complete /opt/cisco/calvados/script/show_tech_aaa: line 27: rse:
command not found .
Compressing show tech output
Show tech output available at /misc/disk1//showtech-aaa-admin-2014-Nov-04.082457.UTC.tgz
Please collect show tech-support ctrace in addition to any sysadmin show-tech-support
collection
++ Show tech end time: 2014-Nov-04.UTC ++
sysadmin-vm:0_RP0#
```

single-connection

To multiplex all TACACS+ requests to this server over a single TCP connection, use the **single-connection** command in TACACS host configuration mode. To disable the single TCP connection for all new sessions that use a separate connection, use the **no** form of this command.

single-connection
no single-connection

Syntax Description

This command has no keywords or arguments.

Command Default

By default, a separate connection is used for each session.

Command Modes

TACACS host configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The **single-connection** command allows the TACACS+ server to handle a greater number of TACACS operations than would be possible if multiple TCP connections were used to send requests to a server.

The TACACS+ server that is being used must support single-connection mode for this to be effective; otherwise, the connection between the network access server and the TACACS+ server locks up or you can receive unauthentic errors.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to configure a single TCP connection to be made with the TACACS+ server (IP address 209.165.200.226) and all authentication, authorization, accounting requests to use this TCP connection. This works only if the TACACS+ server is also configured in single-connection mode. To configure the TACACS+ server in single connection mode, refer to the respective server manual.

```
RP/0/RP0/CPU0:router(config)# tacacs-server host 209.165.200.226
RP/0/RP0/CPU0:router(config-tacacs-host)# single-connection
```

single-connection-idle-timeout

To set the idle timeout value for the single TCP connection to the TACACS+ server, use the **single-connection-idle-timeout** command in *tacacs-server host* configuration mode. To remove the configuration or to disable the idle timeout for the single connection, use the **no** form of this command.

single-connection-idle-timeout *time-in-seconds*

Syntax Description

time-in-seconds	Specifies the single connection timeout value, in seconds. The range is: • 500 to 7200 (prior to Cisco IOS XR Software Release 7.3.2/Release 7.4.1) • 5 to 7200 (from Cisco IOS XR Software Release 7.3.2/Release 7.4.1, and later)

Command Default

Single connection idle timeout is not set, by default.

Command Modes

tacacs-server host

Command History

Release	Modification
Release 7.3.2 Release 7.4.1	This command was modified to change the timeout range.
Release 6.6.3	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
aaa	read, write

Examples

This example shows how to set an idle timeout value of 60 seconds for the single TCP connections to the TACACS+ server:

```
RP/0/RP0/CPU0:router(config)#tacacs-server host 209.165.200.226
RP/0/RP0/CPU0:router(config-tacacs-host)#single-connection-idle-timeout 60
RP/0/RP0/CPU0:router(config-tacacs-host)#commit
```

Related Commands

Command	Description
single-connection, on page 90	Multiplexes all TACACS+ requests to the server over a single TCP connection.

tacacs-server host

To specify a TACACS+ host server, use the **tacacs-server host** command in XR Config mode. To delete the specified name or address, use the **no** form of this command.

tacacs-server host *host-name* [**holddown-time** *time*] [**port** *port-number*] [**timeout** *seconds*] [**key** [{ **0** | **7** }] **auth-key**] [**single-connection**]
[**single-connection-idle-timeout** *time-in-seconds*]
no tacacs-server host *host-name* [*port port-number*]

Syntax Description

host-name	Host or domain name or IP address of the TACACS+ server.
holddown-time *time*	Specifies a duration, in seconds, for which an unresponsive TACACS+ server is to be marked as DOWN. The range is from 0 to 1200. Zero indicates that the hold-down timer feature is disabled.
port *port-number*	(Optional) Specifies a server port number. This option overrides the default, which is port 49. Valid port numbers range from 1 to 65535.
timeout *seconds*	(Optional) Specifies a timeout value that sets the length of time the authentication, authorization, and accounting (AAA) server waits to receive a response from the TACACS+ server. This option overrides the global timeout value set with the **tacacs-server timeout** command for this server only. The valid timeout range is from 1 to 1000 seconds. Default is 5. Note: You can use this parameter only in the config-tacacs-host sub-mode.
key [**0** | **7**] *auth-key*	(Optional) Specifies an authentication and encryption key shared between the AAA server and the TACACS+ server. The TACACS+ packets are encrypted using this key. This key must match the key used by the TACACS+ daemon. Specifying this key overrides the key set by the **tacacs-server key**command for this server only. (Optional) Entering **0** specifies that an unencrypted (clear-text) key follows. (Optional) Entering **7** specifies that an encrypted key follows. The *auth-key* argument specifies the unencrypted key between the AAA server and the TACACS+ server. Note: You can use this parameter only in the config-tacacs-host sub-mode.
single-connection	(Optional) Multiplexes all TACACS+ requests to this server over a single TCP connection. By default, a separate connection is used for each session. Note: You can use this parameter only in the config-tacacs-host sub-mode.

single-connection-idle-timeout *time-in-seconds*	(Optional) Specifies the single connection idle timeout value, in seconds. The range is: • 500 to 7200 (prior to Cisco IOS XR Software Release 7.3.2/Release 7.4.1) • 5 to 7200 (from Cisco IOS XR Software Release 7.3.2/Release 7.4.1, and later)

Command Default

No TACACS+ host is specified.

The *port-name* argument, if not specified, defaults to the standard port 49.

The *seconds* argument, if not specified, defaults to 5 seconds.

Single connection idle timeout is not set, by default.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.4.1	This command was modified to include **holddown-time** option.
Release 7.3.2 Release 7.4.1	This command was modified to change the range for **single-connection-idle-timeout**.
Release 6.6.3	This command was modified to include **single-connection-idle-timeout** option.
Release 6.0	This command was introduced.

Usage Guidelines

You can use multiple **tacacs-server host** commands to specify additional hosts. Cisco IOS XR software searches for hosts in the order in which you specify them.

For details on TACACS+ hold-down timer, see the **holddown-time** command.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to specify a TACACS+ host with the IP address 209.165.200.226:

```
RP/0/RP0/CPU0:router(config)# tacacs-server host 209.165.200.226
RP/0/RP0/CPU0:router(config-tacacs-host)#
```

The following example shows that the default values from the **tacacs-server host** command are displayed from the **show run** command:

```
RP/0/RP0/CPU0:router# show run

Building configuration...
!! Last configuration change at 13:51:56 UTC Mon Nov 14 2005 by lab
!
tacacs-server host 209.165.200.226 port 49
 timeout 5
!
```

The following example shows how to specify that the router consult the TACACS+ server host named host1 on port number 51. The timeout value for requests on this connection is 30 seconds; the encryption key is a_secret.

```
RP/0/RP0/CPU0:router(config)# tacacs-server host host1 port 51
RP/0/RP0/CPU0:router(config-tacacs-host)# timeout 30
RP/0/RP0/CPU0:router(config-tacacs-host)# key a_secret
```

Related Commands

Command	Description
holddown-time (TACACS+), on page 31	Specifies a duration for which an unresponsive TACACS+ server is to be marked as down.
key (TACACS+), on page 35	
single-connection, on page 90	
single-connection-idle-timeout, on page 91	Sets the idle timeout value for the single TCP connection to the TACACS+ server.
timeout (TACACS+), on page 105	

tacacs-server key

To set the authentication encryption key used for all TACACS+ communications between the router and the TACACS+ daemon, use the **tacacs-server key** command in XR Config mode. To disable the key, use the **no** form of this command.

tacacs-server key {**0** *clear-text-key* | **7** *encrypted-keyauth-key*}
no tacacs-server key {**0** *clear-text-key* | **7** *encrypted-keyauth-key*}

Syntax Description

0 *clear-text-key*	Specifies an unencrypted (cleartext) shared key.
7 *encrypted-key*	Specifies an encrypted shared key.
auth-key	Specifies the unencrypted key between the AAA server and the TACACS+ server.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The key name entered must match the key used on the TACACS+ daemon. The key name applies to all servers that have no individual keys specified. All leading spaces are ignored; spaces within and after the key are not. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key.

The key name is valid only when the following guidelines are followed:

- The *clear-text-key* argument must be followed by the **0** keyword.
- The *encrypted-key* argument must be followed by the **7** keyword.

The TACACS server key is used only if no key is configured for an individual TACACS server. Keys configured for an individual TACACS server always override this global key configuration.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example sets the authentication and encryption key to key1:

```
RP/0/RP0/CPU0:router(config)# tacacs-server key key1
```

tacacs-server timeout

To set the interval that the server waits for a server host to reply, use the **tacacs-server timeout** command in XR Config mode. To restore the default, use the **no** form of this command.

tacacs-server timeout *seconds*
no tacacs-server timeout *seconds*

Syntax Description

seconds	Integer that specifies the timeout interval (in seconds) from 1 to 1000.

Command Default

5 seconds

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The TACACS+ server timeout is used only if no timeout is configured for an individual TACACS+ server. Timeout intervals configured for an individual TACACS+ server always override this global timeout configuration.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows the interval timer being changed to 10 seconds:

```
RP/0/RP0/CPU0:router(config)# tacacs-server timeout 10
```

tacacs-server ipv4

To set the Differentiated Services Code Point (DSCP), which is represented by the first six bits in the Type of Service (ToS) byte of the IP header, use the **tacacs-server ipv4** command in XR Config mode.

tacacs-server ipv4 dscp *dscp-value*

Syntax Description

ipv4	Specifies the dscp bit for the IPv4 packets.
dscp	Sets the DSCP in the IP header.
dscp-value	Specifies the options for setting the value of DSCP. The available options are: • <0-63> Differentiated services codepoint value • af11 Match packets with AF11 dscp (001010) • af12 Match packets with AF12 dscp (001100) • af13 Match packets with AF13 dscp (001110) • af21 Match packets with AF21 dscp (010010) • af22 Match packets with AF22 dscp (010100) • af23 Match packets with AF23 dscp (010110) • af31 Match packets with AF31 dscp (011010) • af32 Match packets with AF32 dscp (011100) • af33 Match packets with AF33 dscp (011110) • af41 Match packets with AF41 dscp (100010) • af42 Match packets with AF42 dscp (100100) • af43 Match packets with AF43 dscp (100110) • cs1 Match packets with CS1(precedence 1) dscp (001000) • cs2 Match packets with CS2(precedence 2) dscp (010000) • cs3 Match packets with CS3(precedence 3) dscp (011000) • cs4 Match packets with CS4(precedence 4) dscp (100000) • cs5 Match packets with CS5(precedence 5) dscp (101000) • cs6 Match packets with CS6(precedence 6) dscp (110000) • cs7 Match packets with CS7(precedence 7) dscp (111000) • default Match packets with default dscp (000000) • ef Match packets with EF dscp (101110)

Command Default None

Command Modes XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
aaa	read, write

Examples The following example sets the DSCP value to Assured Forwarding (AF)11:

```
RP/0/RP0/CPU0:router(config)# tacacs-server ipv4 dscp af11
```

tacacs source-interface

To specify the source IP address of a selected interface for all outgoing TACACS+ packets, use the **tacacs source-interface** command in XR Config mode. To disable use of the specified interface IP address, use the **no** form of this command.

tacacs source-interface *type path-id* [**vrf** *vrf-id*]
no tacacs source-interface *type path-id*

Syntax Description

type	Interface type. For more information, use the question mark (?) online help function.
path-id	Physical interface or virtual interface. **Note** Use the **show interfaces** command in XR Config mode to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.
vrf *vrf-id*	Specifies the name of the assigned VRF.

Command Default

If a specific source interface is not configured, or the interface is down or does not have an IP address configured, the system selects an IP address.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **tacacs source-interface** command to set the IP address of the specified interface for all outgoing TACACS+ packets. This address is used as long as the interface is in the *up* state. In this way, the TACACS+ server can use one IP address entry associated with the network access client instead of maintaining a list of all IP addresses.

This command is especially useful in cases where the router has many interfaces and you want to ensure that all TACACS+ packets from a particular router have the same IP address.

When the specified interface does not have an IP address or is in a *down* state, TACACS+ behaves as if no source interface configuration is used.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to set the IP address of the specified interface for all outgoing TACACS+ packets:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# tacacs source-interface TenGigabitEthernet 0/0/0/29 vrf abc
```

task

To add a task ID to a task group, use the **task** command in task group configuration mode. To remove a task ID from a task group, use the **no** form of this command.

task {**read** | **write** | **execute** | **debug**} *taskid-name*
no task {**read** | **write** | **execute** | **debug**} *taskid-name*

Syntax Description

read	Enables read-only privileges for the named task ID.
write	Enables write privileges for the named task ID. The term "write" implies read also.
execute	Enables execute privileges for the named task ID.
debug	Enables debug privileges for the named task ID.
taskid-name	Name of the task ID.

Command Default

No task IDs are assigned to a newly created task group.

Command Modes

Task group configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **task** command in task group configuration mode. To access task group configuration mode, use the **taskgroup** command in global configuration mode.

Task IDs are the base of command authorization. Only users who have the required permissions can execute a particular command on the router. To execute a command, the user must be part of a user group that consists of task group(s) that includes required task IDs and privileges. Cisco IOS XR software supports multiple task IDs. For example, **aaa**, **config-services**, **crypto**, **system**, and so on. To see the list of task IDs available for the user, use the **show user tasks** command.

Likewise, all commands are associated with one or more task IDs, and their corresponding operations (such as **read**, **write**, **execute**, and **debug**) that denote the permissions required to execute those commands. You can use the **describe** command to know the task ID and permissions that are required to execute a particular command.

For example, the following output shows that the user needs **aaa** task ID with **read** and **write** permission to execute the **show run aaa** command. So, users can execute this command if they belong to a user group associated with a task group that includes this **aaa** task ID having read and write privileges.

```
Router# describe show run aaa
The command is defined in aaa_cmds.parser

User needs ALL of the following taskids:

    aaa (READ WRITE) ----------->

It will take the following actions:
```

```
Wed Mar 16 07:58:01.451 UTC
  Spawn the process:
    nvgen "-c" "-q" "gl/aaa/"
Router#
```

Root users (users in **root-lr** or **root-system** user group) have all task IDs, and hence will be able to execute all commands. Also, certain commands might not require any task ID as such to execute it. So, all users will have permission to execute such commands. If you do not have the required permission to execute a command, the command authorization fails. If the user group assignment is preventing you from using any command, contact your AAA administrator for assistance.

A few other examples that describe the commands to list the task ID:

```
Router#describe show interfaces
The command is defined in show_interface.parser

show_interface.parser
User needs ALL of the following taskids:

   interface (READ)---------->

It will take the following actions:
Thu Mar 17 06:42:08.264 UTC
  Spawn the process:
    show_interface "-a"
Router#


Router(config)#describe ssh server
The command is defined in ssh.parser

ssh.parser
User needs ALL of the following taskids:

   crypto (READ WRITE) ---------->

It will take the following actions:
  Create/Set the configuration item:
       Path: gl/crypto/ssh/server/sshd/vrf/default
       Value: packed[ 0x1 <string> <string> ]
Router(config)#
```

For more details, see *Configuring AAA Services* chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to enable execute privileges for the config-services task ID and associate that task ID with the task group named taskgroup1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# taskgroup taskgroup1
RP/0/RP0/CPU0:router(config-tg)# task execute config-services
```

taskgroup

To configure a task group to be associated with a set of task IDs, and to enter task group configuration mode, use the **taskgroup** command in XR Config mode. To delete a task group, use the **no** form of this command.

taskgroup *taskgroup-name* [{**description** *string* | **task** {**read** | **write** | **execute** | **debug**} *taskid-name* | **inherit taskgroup** *taskgroup-name*}]
no taskgroup *taskgroup-name*

Syntax Description

taskgroup-name	Name of a particular task group.
description	(Optional) Enables you to create a description for the named task group.
string	(Optional) Character string used for the task group description.
task	(Optional) Specifies that a task ID is to be associated with the named task group.
read	(Optional) Specifies that the named task ID permits read access only.
write	(Optional) Specifies that the named task ID permits read and write access only.
execute	(Optional) Specifies that the named task ID permits execute access.
debug	(Optional) Specifies that the named task ID permits debug access only.
taskid-name	(Optional) Name of a task: the task ID.
inherit taskgroup	(Optional) Copies permissions from the named task group.
taskgroup-name	(Optional) Name of the task group from which permissions are to be inherited.

Command Default

Five predefined user groups are available by default.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Task groups are configured with a set of task IDs for each action type. Deleting a task group that is still referenced in the system results in a warning and rejection of the deletion. For more details on task IDs, see the Usage Guidelines section of the **task** command.

You can use the **show user group** command in XR Config mode to know the group(s) that the current user is part of. Similarly, you can use the **show user all** to know the group or task information (such as username, groups, authentication method, task IDs, and so on) of the current user.

From global configuration mode, you can display all the configured task groups. However, you cannot display all the configured task groups in taskgroup configuration mode.

Entering the **taskgroup** command with no keywords or arguments enters task group configuration mode, in which you can use the **description**, **inherit**, **show**, and **task** commands.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example assigns read bgp permission to the task group named alpha:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# taskgroup alpha
RP/0/RP0/CPU0:router(config-tg)# task read bgp
```

timeout (TACACS+)

To specify a timeout value that sets the length of time the authentication, authorization, and accounting (AAA) server waits to receive a response from the TACACS+ server, use the **timeout** (TACACS+) command in TACACS host configuration mode. To disable this command and return to the default timeout value of 5 seconds, use the **no** form of this command.

timeout *seconds*
no timeout *seconds*

Syntax Description

seconds	Timeout value (in seconds). The range is from 1 to 1000. If no timeout is specified, the global value is used.

Command Default

seconds: 5

Command Modes

TACACS host configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The **timeout** (TACACS+) command overrides the global timeout value set with the **tacacs-server timeout** command for this server only.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to set the number of seconds for the timeout value:

```
RP/0/RP0/CPU0:router(config)# tacacs-server host 209.165.200.226
RP/0/RP0/CPU0:router(config-tacacs-host)# timeout 500
```

timeout login response

To set the interval that the server waits for a reply to a login, use the **timeout login response** command in line template configuration mode. To restore the default, use the **no** form of this command.

timeout login response *seconds*
no timeout login response *seconds*

Syntax Description

seconds	Integer that specifies the timeout interval (in seconds) from 0 to 300.

Command Default

seconds: 30

Command Modes

Line template configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **timeout login response** command in line template configuration mode to set the timeout value. This timeout value applies to all terminal lines to which the entered line template is applied. This timeout value cannot be applied to line console. After the timeout value has expired, the user is prompted again. The retry is allowed three times.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to change the interval timer to 20 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# line template alpha
RP/0/RP0/CPU0:router(config-line)# timeout login response 20
```

usergroup

To configure a user group and associate it with a set of task groups, and to enter user group configuration mode, use the **usergroup** command in XR Config mode. To delete a user group, or to delete a task-group association with the specified user group, use the **no** form of this command.

usergroup *usergroup-name*
no usergroup *usergroup-name*

Syntax Description

usergroup-name	Name of the user group. The *usergroup-name* argument can be only one word. Spaces and quotation marks are not allowed.

Command Default

Five predefined user groups are available by default.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

User groups are configured with the command parameters for a set of users, such as task groups. You can remove specific user groups by using the **no** form of the **usergroup** command. You can remove the user group itself by using the **no** form of the command without giving any parameters. Deleting a user group that is still referenced in the system results in a warning and a rejection of the deletion.

Use the inherit usergroup, on page 34 command to copy permissions from other user groups. The user group is inherited by the parent group and forms a union of all task IDs specified in those groups. Circular inclusions are detected and rejected. User groups cannot inherit properties from predefined groups, such as root-system and owner-sdr.

From global configuration mode, you can display all the configured user groups. However, you cannot display all the configured user groups in usergroup configuration mode.

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows how to add permissions from the user group beta to the user group alpha:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# usergroup alpha
RP/0/RP0/CPU0:router(config-ug)# inherit usergroup beta
```

username

To configure a new user with a username, establish a password, grant permissions for the user, and to enter username configuration mode, use the **username** command in XR Config mode or System Admin Config mode. To delete a user from the database, use the **no** form of this command.

username *name* [{ **group** *name* | **policy** *name* | [**password-policy** *name*] { **password** | **masked-password** } [*type*] *password* | { **secret** | **masked-secret** } [{ *type* | **0** [**enc-type** *type*] *secret* }] }]
no username *name* [{ **group** *name* | **policy** | **password** | **masked-password** | **secret** | **masked-secret** | **password-policy** *name* [**masked-password** [*type*] *password*] }]

Syntax Description

name	Name of the user. The *name* argument can be only one word. Spaces and quotation marks are not allowed. The allowed range for a user-defined username is 2-253 characters.
group *name*	Enables a user to be associated with a user group, as defined with the **usergroup** command.
policy *name*	Configures a password policy that is common to user password and secret.
password-policy *name*	(Optional) Specifies the password policy for cleartext and Type 7 password authentication.
password	Enables a password to be created for the specified user.
masked-password	Enables a password to be created for the specified user. When you key in the password, it is not visible on the screen.

type password	Specifies the password type and the password to be keyed in. Enter 0 or 7 for the *type* argument. 0 specifies a cleartext password, and 7 specifies a Type 7 encrypted password. If Type 7 encryption is enabled with the **password** keyword, the password is not visible to the user. The password can be up to 253 characters in length. (Optional) *type* argument
secret	Enables a secret to be created for the specified user.
masked-secret	Enables a secret to be created for the specified user. When you key in the secret, it is not visible on the screen.
type secret	Specifies the secret type and the secret to be keyed in. Enter 0, or enter 5, 8, 9, or 10, for the *type* argument. Details: • 0 specifies a cleartext secret that will be encrypted for use. • 5 specifies a Type 5 password that uses MD5 hashing algorithm. • 8 specifies a Type 8 password that uses SHA256 hashing algorithm. • 9 specifies a Type 9 password that uses scrypthashing algorithm. • 10 specifies a Type 10 password that uses SHA512 hashing algorithm. (Optional) *type* argument.

0 enc-type *type secret*	Specifies that you enter a cleartext secret to be encrypted by a specified encryption method. • 0 specifies that you should enter a cleartext secret. • **enc-type** specifies that you enter 5, 8, 9, or 10, for the *type* argument. • Enter the cleartext secret for the *secret* argument. (Optional) **enc-type** *type* keyword-argument combination.

Command Default

No usernames are defined in the system.

Command Modes

XR Config mode

System Admin Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.
Release 6.2.1	Added support for **password-policy**, as part of AAA password security for FIPS compliance.
Release 7.0.1	Added the support for Type 8 (SHA256), Type 9 (scrypt) and Type 10 (SHA512) for **secret** configuration.
Release 7.2.1	Added the support for **policy** option to configure policy common to user password and secret.
Release 7.3.1	Password Masking feature options (**masked-password** and **masked-secret**) were added. When you key in a password or secret, it is not displayed on the screen

Usage Guidelines

Note

- A user is never allowed to have cisco-support privileges as the only group.
- From Release 7.0.1 and later, Type 10 (SHA512) is applied as the default type for the **secret** configuration. Prior to this, Type 5 (MD5) was the default one.

Use the **username** command to identify the user and enter username configuration mode. Password and user group assignments can be made from either XR Config mode or username configuration submode. Permissions (task IDs) are assigned by associating the user with one or more defined user groups.

From XR Config mode, you can display all the configured usernames. However, you cannot display all the configured usernames in username configuration mode.

Each user is identified by a username that is unique across the administrative domain. Each user should be made a member of at least one user group. Deleting a user group may orphan the users associated with that group. The AAA server authenticates orphaned users, but most commands are not authorized.

The **username** command is associated with a particular user for local login authentication by default. Alternatively, a user and password can be configured in the database of the TACACS+ server for TACACS+ login authentication. For more information, see the description of the aaa authentication (XR-VM), on page 8 command.

The predefined group root-system may be specified only by root-system users while administration is configured.

Note To enable the local networking device to respond to remote Challenge Handshake Authentication Protocol (CHAP) challenges, one **username** command entry must be the same as the hostname entry that has already been assigned to the other networking device.

For more details on defining a password policy, refer **aaa password-policy** command. The AAA password security policy feature works as such for Cisco IOS XR platforms. Whereas, it is supported only on XR VM, for Cisco IOS XR 64 bit platforms.

The following are password masking guidelines for various command forms:

- **username** *name* **password** *type password*

  username *name* **masked-password** *type password*

  Enter 0 or 7 for the *type* argument. *0* specifies a cleartext password, and *7* specifies a Type 7 encrypted password.

- **secret** *type secret*

  masked-secret *type secret*

  Enter 0, or enter 5, 8, 9, or 10, for the *type* argument. 0 specifies a cleartext secret, and 5, 8, 9, and 10 specify a Type 5, Type 8, Type 9, and Type 10 secret, respectively.

- **secret 0 enc-type** *type secret*

  masked-secret 0 enc-type *type secret*

Enter 5, 8, 9, or 10, for the *type* argument.

- **masked-password** *type password*

  masked-secret *type secret*

  After specifying the password encryption type, press **Enter** or **return** on your keyboard. The password/secret option appears in the next line. Example:

```
Router(config)# masked-secret 10

Enter secret:
Re-enter secret:
```

Task ID

Task ID	Operations
aaa	read, write

Examples

The following example shows the commands available after executing the **username** command:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# username user1
RP/0/RP0/CPU0:router(config-un)# ?
```

clear	Clear the uncommitted configuration
commit	Commit the configuration changes to running
describe	Describe a command without taking real actions
do	Run an exec command
exit	Exit from this submode
group	User group in which this user will be a member of
no	Negate a command or set its defaults
password	Specify the password for the user
policy	Specify the policy common to password and secret for the user
pwd	Commands used to reach current submode
root	Exit to the XR Config mode
secret	Specify the secure password for the user
show	Show contents of configuration

```
RP/0/RP0/CPU0:router(config-un)#
```

The following example shows how to establish the clear-text password *password1* for the user name *user1*:

```
RP/0/RP0/CPU0:router# configure
```

```
RP/0/RP0/CPU0:router(config)# username user1
RP/0/RP0/CPU0:router(config-un)# password 0 password1
```

This example shows how to apply a AAA password policy for a user:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# username user1 password-policy test-policy password abc
```

This example shows how to apply a password policy for the user secret:

```
Router#configure
Router(config)#username user1
Router(config-un)#policy test-policy1
Router(config-un)#secret 10
$6$dmwuW0Ajicf98W0.$y/vzynWF1/OcGxwBwHs79VAy5ZZLhoHd7TicR4mOo8IIVriYCGAKW0A.w1JvTPO7IbZry.DxHrE3SN2BBzBJe0
Router(config-un)#commit
```

The following example shows how to configure a Type 8 (SHA256) password for the user, *user8*. You can also see the examples and usage of the secret, on page 51 command.

You can specify Type as '8' under the **secret** keyword, to explicitly configure Type 8 password.

```
Router#configure
Router(config)#username user8 secret 8
$8$ZYKGl1dZIw73Dl$IUWJOqTLoMyExhsNKoL5vMtvCOYguM5ajXf4uGeQj6I
Router(config-un)#commit
```

This example shows how to configure Type 9 password:

```
Router#configure
Router(config)#username user9 secret 9
$9$/rIQL1B3rplRBL$oS2fLWKFYH6B/kApxkkXmIqbPAHpRZkPEoh3WqGbvwQ
Router(config-un)#commit
```

Similarly, this example shows how to configure Type 10 password :

```
Router#configure
Router(config)#username user10 secret 10
$6$9UvJidvsTEqgkAPU$3CL1Ei/F.E4v/Hi.UaqLwX8UsSEr9ApG6c5pzhMJmZtgW4jObAQ7meAwyhu5VM/aRFJqe/jxZG17h6xPrvJWf1
Router(config-un)#commit
```

This example shows how to specify the Type 10 password in System Admin VM:

```
Router#admin
sysadmin-vm:0_RP0# configure
sysadmin-vm:0_RP0(config)# aaa authentication users user user10 password testpassword
sysadmin-vm:0_RP0(config)# commit
```

Password Masking Examples

The following example shows how to enable password masking for a cleartext password entry:

In this example, for user us3, a cleartext password is entered.

```
Router(config)# username us3 masked-password 0


Enter password:
```

```
Re-enter password:

Router(config)#commit
```

In the **show** command output, you can see the encrypted password:

```
Router# show run aaa
..

username us3
 password 7 105A1D0D
```

The encrypted password 105A1D0D is entered in the **Enter password:** and **Re-enter password:** fields, for Type 7 password encryption:

```
Router(config)# username us3 masked-password 7

Enter password:
Re-enter password:


Router(config)#commit
```

If there is a password mismatch between the two entries, an error message is displayed.

The following example shows how to enable password masking for a AAA password policy:

In this example, for user us6, a cleartext password is entered.

```
Router(config)# aaa password-policy security
Router(config)# username us6 password-policy security masked-password 0

Enter password:
Re-enter password:

Router(config)#commit
```

In the **show** command output, you can see the encrypted password.

```
Router# show run aaa
..

aaa password-policy security
..
username us6
 password-policy security password 7 0835585A
```

The encrypted password 0835585A is entered in the **Enter password:** and **Re-enter password:** fields for Type 7 password encryption.

```
Router(config)# username us6 password-policy test-policy masked-password 7

Enter password:
Re-enter password:


Router(config)#commit
```

users group

To associate a user group and its privileges with a line, use the **users group** command in line template configuration mode. To delete a user group association with a line, use the **no** form of this command.

users group {*usergroup-name* | **cisco-support** | **netadmin** | **operator** | **root-lr** | **root-system** | **sysadmin**}

no users group {*usergroup-name* | **cisco-support** | **netadmin** | **operator** | **root-lr** | **root-system** | **serviceadmin** | **sysadmin**}

Syntax Description

usergroup-name	Name of the user group. The *usergroup-name* argument can be only one word. Spaces and quotation marks are not allowed.
cisco-support	Specifies that users logging in through the line are given Cisco support personnel privileges.
netadmin	Specifies that users logging in through the line are given network administrator privileges.
operator	Specifies that users logging in through the line are given operator privileges.
root-lr	Specifies that users logging in through the line are given root logical router (LR) privileges.
root-system	Specifies that users logging in through the line are given root system privileges.
serviceadmin	Specifies that users logging in through the line are given service administrator group privileges.
sysadmin	Specifies that users logging in through the line are given system administrator privileges.

Command Default

None

Command Modes

Line template configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **users group** command to enable a user group and its privileges to be associated with a line, meaning that users logging in through the line are given the privileges of the particular user group.

Task ID

Task ID	Operations
aaa	read, write

Examples

In the following example, if a vty-pool is created with line template *vty*, users logging in through vty are given operator privileges:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa authen login vty-authen line
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router(config)# line template vty
RP/0/RP0/CPU0:router(config-line)# users group operator
RP/0/RP0/CPU0:router(config-line)# login authentication
```

Keychain Management Commands

This module describes the commands used to configure keychain management.

For detailed information about keychain management concepts, configuration tasks, and examples, see the Implementing Keychain Management chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Note Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

accept-lifetime

To set the time period during which the authentication key on a keychain is received as valid, use the **accept-lifetime** command in key configuration mode. To revert to the default value, use the **no** form of this command.

accept-lifetime *start-time* [{**duration** *duration value* | **infinite***end-time*}]
no accept-lifetime *start-time* [{**duration** *duration value* | **infinite***end-time*}]

Syntax Description

start-time	Start time, in *hh:mm:ss day month year* format, in which the key becomes valid. The range is from 0:0:0 to 23:59:59. The range for the number of days of the month is from 1 to 31. The range for the years is from 1993 to 2035.
duration *duration value*	(Optional) Determines the lifetime of the key in seconds. The range is from 1-2147483646.
infinite	(Optional) Specifies that the key never expires after it becomes valid.
end-time	(Optional) Time, in *hh:mm:ss day month year* format, after which the key expires. The range is from 0:0:0 to 23:59:59.

Command Default

None

Command Modes

Key configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to use the **accept-lifetime** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# key chain isis-keys
RP/0/RP0/CPU0:router(config-isis-keys)# key 8
RP/0/RP0/CPU0:router(config-isis-keys-0x8)# accept-lifetime 1:00:00 June 29 2006 infinite
```

accept-tolerance

To specify the tolerance or acceptance limit, in seconds, for an accept key that is used by a peer, use the **accept-tolerance** command in keychain configuration mode. To disable this feature, use the **no** form of this command.

accept-tolerance [{*value* | **infinite**}]
no accept-tolerance [{*value* | **infinite**}]

Syntax Description

value	(Optional) Tolerance range, in seconds. The range is from 1 to 8640000.
infinite	(Optional) Specifies that the tolerance specification is infinite. The accept key never expires. The tolerance limit of infinite indicates that an accept key is always acceptable and validated when used by a peer.

Command Default

The default value is 0, which is no tolerance.

Command Modes

Keychain configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

If you do not configure the **accept-tolerance** command, the tolerance value is set to zero.

Even though the key is outside the active lifetime, the key is deemed acceptable as long as it is within the tolerance limit (for example, either prior to the start of the lifetime, or after the end of the lifetime).

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to use the **accept-tolerance** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# key chain isis-keys
RP/0/RP0/CPU0:router(config-isis-keys)# accept-tolerance infinite
```

cryptographic-algorithm

To apply the cryptographic algorithm to the packets using the key string configured for the key ID, use the **cryptographic-algorithm** command in keychain-key configuration mode. To disable this feature, use the **no** form of this command.

cryptographic-algorithm [{ **HMAC-MD5** | **HMAC-SHA1-12** | **HMAC-SHA1-20** | **MD5** | **SHA-1** | **HMAC-SHA-256** | **HMAC-SHA1-96** | **AES-128-CMAC-96** }]

Syntax Description

HMAC-MD5	Configures HMAC-MD5 as a cryptographic algorithm with a digest size of 16 bytes.
HMAC-SHA1-12	Configures HMAC-SHA1-12 as a cryptographic algorithm with a digest size of 12 bytes.
HMAC-SHA1-20	Configures HMAC-SHA1-20 as a cryptographic algorithm with a digest size of 20 bytes.
MD5	Configures MD5 as a cryptographic algorithm with a digest size of 16 bytes.
SHA-1	Configures SHA-1-20 as a cryptographic algorithm with a digest size of 20 bytes.
HMAC-SHA-256	Configures HMAC-SHA-256 as a cryptographic algorithm with a digest size of 32 bytes.
HMAC-SHA1-96	Configures HMAC-SHA1-96 as a cryptographic algorithm with a digest size of 12 bytes.
AES-128-CMAC-96	Configures AES-128-CMAC as a cryptographic algorithm with a digest size of 12 bytes.

Command Default

No default behavior or values

Command Modes

Keychain-key configuration

Command History

Release	Modification
Release 6.0	This command was introduced.
Release 6.5.1	Support for the following algorithms are added: • HMAC-SHA-256 • HMAC-SHA1-96 • AES-128-CMAC-96

Usage Guidelines

If you do not specify the cryptographic algorithm, MAC computation and API verification would be invalid.

These protocols support the following cryptographic algorithms:

- Border Gateway Protocol (BGP) supports only HMAC-MD5, HMAC-SHA1-12, AES-128-CMAC-96 and HMAC-SHA1-96.

- Intermediate System-to-Intermediate System (IS-IS) supports HMAC-MD5, SHA-1, MD5, AES-128-CMAC-96, HMAC-SHA-256, HMAC-SHA1-12, HMAC-SHA1-20, and HMAC-SHA1-96.
- Open Shortest Path First (OSPF) supports MD5, HMAC-MD5, HMAC-SHA-256, HMAC-SHA1-12, HMAC-SHA1-20, and HMAC-SHA1-96.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to use the **cryptographic-algorithm** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# key chain isis-keys
RP/0/RP0/CPU0:router(config-isis-keys)# key 8
RP/0/RP0/CPU0:router(config-isis-keys-0x8)# cryptographic-algorithm HMAC-MD5
```

key (key chain)

To create or modify a keychain key, use the **key** command in keychain-key configuration mode. To disable this feature, use the **no** form of this command.

key *key-id*
no key *key-id*

Syntax Description

key-id	48-bit integer key identifier of from 0 to 281474976710655.

Command Default

No default behavior or values

Command Modes

Keychain-key configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

For a Border Gateway Protocol (BGP) keychain configuration, the range for the *key-id* argument must be from 0 to 63. If the range is above the value of 63, the BGP keychain operation is rejected.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to use the **key** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# key chain isis-keys
RP/0/RP0/CPU0:router(config-isis-keys)# key 8
RP/0/RP0/CPU0:router(config-isis-keys-0x8)#
```

key chain (key chain)

To create or modify a keychain, use the **key chain** command . To disable this feature, use the **no** form of this command.

key chain *key-chain-name*
no key chain *key-chain-name*

Syntax Description

key-chain-name	Specifies the name of the keychain. The maximum number of characters is 48.

Command Default

No default behavior or values

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

You can configure a keychain for Border Gateway Protocol (BGP) as a neighbor, session group, or neighbor group. BGP can use the keychain to implement a hitless key rollover for authentication.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows that the name of the keychain isis-keys is for the **key chain** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# key chain isis-keys
RP/0/RP0/CPU0:router(config-isis-keys)#
```

key-string (keychain)

To specify the text string for the key, use the **key-string** command in keychain-key configuration mode. To disable this feature, use the **no** form of this command.

key-string [{**clear** | **password**}] *key-string-text*
no key-string [{**clear** | **password**}] *key-string-text*

Syntax Description

clear	Specifies the key string in clear-text form.
password	Specifies the key in encrypted form.
key-string-text	Text string for the key, which is encrypted by the parser process before being saved to the configuration. The text string has the following character limitations: • Plain-text key strings—Minimum of 1 character and a maximum of 32. • Encrypted key strings—Minimum of 4 characters and no maximum.

Command Default

The default value is clear.

Command Modes

Keychain-key configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

For an encrypted password to be valid, the following statements must be true:

- String must contain an even number of characters, with a minimum of four.
- The first two characters in the password string must be decimal numbers and the rest must be hexadecimals.
- The first two digits must not be a number greater than 53.

Either of the following examples would be valid encrypted passwords:

1234abcd

or

50aefd

From Cisco IOS XR Software Release 7.1.2, Release 7.2.1 and later, if you are using any **HMAC-SHA** algorithm for a session, then you must ensure that the configured *key-string* has a minimum length of 14 characters. Otherwise, the session goes down. This guideline is applicable only for FIPS mode.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to use the **keystring** command:

```
RP/0/RP0/CPU0:router:# configure
RP/0/RP0/CPU0:router(config)# key chain isis-keys
RP/0/RP0/CPU0:router(config-isis-keys)# key 8
RP/0/RP0/CPU0:router(config-isis-keys-0x8)# key-string password 850aefd
```

send-lifetime

To send the valid key and to authenticate information from the local host to the peer, use the **send-lifetime** command in keychain-key configuration mode. To disable this feature, use the **no** form of this command.

send-lifetime *start-time* [{**duration** *duration value* | **infinite***end-time*}]
no send-lifetime *start-time* [{**duration** *duration value* | **infinite***end-time*}]

Syntax Description

start-time	Start time, in *hh:mm:ss day month year* format, in which the key becomes valid. The range is from 0:0:0 to 23:59:59. The range for the number of days of the month to start is from 1 to 31. The range for the years is from 1993 to 2035.
duration *duration value*	(Optional) Determines the lifetime of the key in seconds.
infinite	(Optional) Specifies that the key never expires once it becomes valid.
end-time	(Optional) Time, in *hh:mm:ss day month year* format, after which the key expires. The range is from 0:0:0 to 23:59:59

Command Default

No default behavior or values

Command Modes

Keychain-key configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to use the **send-lifetime** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# key chain isis-keys
RP/0/RP0/CPU0:router(config-isis-keys)# key 8
RP/0/RP0/CPU0:router(config-isis-keys-0x8)# send-lifetime 1:00:00 June 29 2006 infinite
```

show key chain

To display the keychain, use the **show key chain** command.

show key chain *key-chain-name*

Syntax Description

key-chain-name	Names of the keys in the specified keychain. The maximum number of characters is 32.

Command Default

If the command is used without any parameters, then it lists out all the key chains.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
system	read

Examples

When a secure key storage becomes available, it is desirable for keychain management to alternatively prompt you for a primary password and display the key label after decryption. The following example displays only the encrypted key label for the **show key chain** command:

```
RP/0/RP0/CPU0:router# show key chain isis-keys

Key-chain: isis-keys/ -

accept-tolerance -- infinite
Key 8 -- text "8"
  cryptographic-algorithm -- MD5
  Send lifetime:   01:00:00, 29 Jun 2006 - Always valid  [Valid now]
  Accept lifetime: 01:00:00, 29 Jun 2006 - Always valid [Valid now]
```

Management Plane Protection Commands

This module describes the commands used to configure management plane protection (MPP).

For detailed information about keychain management concepts, configuration tasks, and examples, see the Implementing Management Plane Protection chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Note Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

address ipv4 (MPP)

To configure the peer IPv4 or IPv6 address in which management traffic is allowed on the interface, use the **address ipv4**command in interface peer configuration mode. To remove the IP address that was previously configured on this interface, use the **no** form of this command.

address {**ipv4** | **ipv6**}
peer-ip-address
|*peer-ip-address*/*length*
no address {**ipv4** | **ipv6**}
peer-ip-address
| *peer-ip-address*/*length*
address ipv4 {*peer-ip-address* | *peer-ip-address*/ *length*}
no address ipv4 {*peer-ip-address* | *peer-ip-address*/ *length*}

Syntax Description

peer-ip-address	(Required) Peer IPv4 or IPv6 address in which management traffic is allowed on the interface. This address can effectively be the source address of the management traffic that is coming in on the configured interface.
peer ip-address/length	(Required) Prefix of the peer IP address and IPv4 address. or IPv6 format: • IPv4—*A.B.C.D/length* • IPv6—*X.X:X.X*

Command Default

If no specific peer is configured, all peers are allowed.

Command Modes

Interface peer configuration

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to configure the peer IPv4 address 10.1.0.0, with a prefix of 16 for traffic management:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)# management-plane
RP/0/RP0/CPU0:router(config-mpp)# inband
```

```
RP/0/RP0/CPU0:router(config-mpp-inband)# interface all
RP/0/RP0/CPU0:router(config-mpp-inbandoutband-all)# allow all peer
RP/0/RP0/CPU0:router(config-telnettftp-peer)# address ipv4 10.1.0.0/16
```

address ipv6 (MPP)

To configure the peer IPv6 address in which management traffic is allowed on the interface, use the **address ipv6** command in interface peer configuration mode. To remove the IP address that was previously configured on this interface, use the **no** form of this command.

address ipv6 {*peer-ip-address* | *peer-ip-address*/*length*}

Syntax Description

peer-ip-address	Peer IPv6 address in which management traffic is allowed on the interface. This address can effectively be the source address of the management traffic that is coming in on the configured interface.
s*peer ip-address/length*	Prefix of the peer IPv6 address.

Command Default

If no specific peer is configured, all peers are allowed.

Command Modes

Interface peer configuration

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to configure the peer IPv6 address 33::33 for management traffic:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)# management-plane
RP/0/RP0/CPU0:router(config-mpp)# inband
RP/0/RP0/CPU0:router(config-mpp-outband)# interface GigabitEthernet 0/1/1/2
RP/0/RP0/CPU0:router(config-mpp-outband-GigabitEthernet0_1_1_2)# allow TFTP peer
RP/0/RP0/CPU0:router(config-tftp-peer)# address ipv6 33::33
```

allow (MPP)

To configure an interface as an inband interface to allow all peer addresses for a specified protocol or all protocols, use the **allow** command in management plane protection inband interface configuration mode.

To disallow a protocol on an interface, use the **no** form of this command.

allow {*protocol* | **all**} [**peer**]
no allow {*protocol* | **all**} [**peer**]

Syntax Description

protocol	Interface configured to allow peer-filtering for the following specified protocol's traffic: • Netconf • SNMP (also versions) • Secure Shell (v1 and v2) • TFTP • Telnet • XML
all	Configures the interface to allow peer-filtering for all the management traffic that is specified in the list of protocols.
peer	(Optional) Configures the peer address on the interface. Peer refers to the neighboring router interface in which traffic might arrive to the main router.

Command Default

By default, no management protocol is allowed on any interface except the management interfaces.

Command Modes

Management plane protection inband interface configuration

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Usage Guidelines

If you permit or allow a specific protocol to an interface, traffic is allowed only for that protocol, and all other management traffic is dropped.

The IOS XR XML API provides a programmatic interface to the router for use by external management applications. This interface provides a mechanism for router configuration and monitoring utilizing XML formatted request and response streams. As one of the management services, XML should be capable of applying MPP. To secure XML MPP data, XML keyword has been added to the command.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to configure all management protocols for all inband interfaces:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)# management-plane
RP/0/RP0/CPU0:router(config-mpp)# inband
RP/0/RP0/CPU0:router(config-mpp-inband)# interface all
RP/0/RP0/CPU0:router(config-mpp-inband-all)# allow all
```

The following example shows how to configure MPP support on an XML peer in-band interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)# management-plane
RP/0/RP0/CPU0:router(config-ctrl-mpp)# inband interface all allow xml peer address ipv4
172.10.10.1
```

control-plane

To enter the control plane configuration mode, use the **control-plane** command. To disable all the configurations under control plane mode, use the **no** form of this command.

control-plane
no control-plane

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Usage Guidelines

Use the **control-plane** command to enter control plane configuration mode.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to enter control plane configuration mode using the **control-plane** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)#
```

inband

To configure an inband interface and to enter management plane protection inband configuration mode, use the **inband** command in management plane protection configuration mode. To disable all configurations under inband configuration mode, use the **no** form of this command.

inband
no inband

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Management plane protection inband configuration

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Usage Guidelines

Use the **inband** command to enter management plane protection inband configuration mode.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to enter management plane protection inband configuration mode using the **inband** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)# management-plane
RP/0/RP0/CPU0:router(config-mpp)# inband
RP/0/RP0/CPU0:router(config-mpp-inband)#
```

interface (MPP)

To configure a specific interface or all interfaces as an inband interface, use the **interface** command in management plane protection inband configuration mode.

To disable all the configurations under an interface mode, use the **no** form of this command.

interface {*type* *interface-path-id* | **all**}
no interface {*type* *interface-path-id* | **all**}

Syntax Description

type	Interface type. For more information, use the question mark (?) online help function.
interface-path-id	Virtual interface instance. Number range varies depending on interface type. **Note** Use the **show interfaces** command in EXEC mode to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.
all	Configures all interfaces to allow for management traffic.

Command Default

None

Command Modes

Management plane protection inband configuration

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Usage Guidelines

Use the **interface** command to enter management plane protection inband interface configuration mode.

For the *instance* argument, you cannot configure Management Ethernet interfaces as inband interfaces.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to configure all inband interfaces for MPP:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)# management-plane
RP/0/RP0/CPU0:router(config-mpp)# inband
RP/0/RP0/CPU0:router(config-mpp-inband)# interface all
RP/0/RP0/CPU0:router(config-mpp-inband-all)#
```

management-plane

To configure management plane protection to allow and disallow protocols, use the **management-plane** command in control plane configuration mode. To disable all configurations under management-plane mode, use the **no** form of this command.

management-plane
no management-plane

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Control plane configuration

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Usage Guidelines

Use the **management-plane** command to enter the management plane protection configuration mode.

Task ID

Task ID	Operations
system	read, write

Examples

The following example shows how to enter management plane protection configuration mode using the **management-plane** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# control-plane
RP/0/RP0/CPU0:router(config-ctrl)# management-plane
RP/0/RP0/CPU0:router(config-mpp)#
```

show mgmt-plane

To display information about the management plane such as type of interface and protocols enabled on the interface, use the **show mgmt-plane** command.

show mgmt-plane [{**inband**}] [{**interface** *type interface-path-id* | **vrf**}]

Syntax Description

inband	(Optional) Displays the inband management interface configurations that are the interfaces that process management packets as well as data-forwarding packets. An inband management interface is also called a *shared management interface*.
interface	(Optional) Displays all the protocols that are allowed in the specified interface.
type	Interface type. For more information, use the question mark (?) online help function.
interface-path-id	Interface instance. Number range varies depending on interface type. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0.1	This command was introduced.

Task ID

Task ID	Operations
system	read

Examples

The following sample output displays all the interfaces that are configured as inband under MPP:

```
RP/0/RP0/CPU0:router# show mgmt-plane

Management Plane Protection

inband interfaces
----------------------

interface - TenGigabitEthernet0_1_1_0
        ssh configured -
                All peers allowed
        telnet configured -
                peer v4 allowed - 10.1.0.0/16
```

```
        all configured -
                All peers allowed
interface - TenGigabitEthernet0_1_1_0
        telnet configured -
                peer v4 allowed - 10.1.0.0/16
```

Secure Shell Commands

This module describes the Cisco IOS XR software commands used to configure Secure Shell (SSH).

For detailed information about SSH concepts, configuration tasks, and examples, see the Implementing Secure Shell chapter in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Note Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

clear ssh

To terminate an incoming or outgoing Secure Shell (SSH) connection, use the **clear ssh** command.

clear ssh {*session-id* | **outgoing** *session-id*}

Syntax Description

session-id	Session ID number of an incoming connection as displayed in the **show ssh** command output. Range is from 0 to 1024.
outgoing *session-id*	Specifies the session ID number of an outgoing connection as displayed in the **show ssh** command output. Range is from 1 to 10.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **clear ssh** command to disconnect incoming or outgoing SSH connections. Incoming connections are managed by the SSH server running on the local networking device. Outgoing connections are initiated from the local networking device.

To display the session ID for a connection, use the **show ssh** command.

Task ID

Task ID	Operations
crypto	execute

Examples

In the following example, the **show ssh** command is used to display all incoming and outgoing connections to the router. The **clear ssh** command is then used to terminate the incoming session with the ID number 0.

```
RP/0/RP0/CPU0:router# show ssh

SSH version: Cisco-2.0
session     pty  location   state       userid     host        ver
-------------------------------------------------------------------
Incoming sessions
0           vty0 0/33/1  SESSION_OPEN   cisco   172.19.72.182   v2
1           vty1 0/33/1  SESSION_OPEN   cisco   172.18.0.5      v2
2           vty2 0/33/1  SESSION_OPEN   cisco   172.20.10.3     v1
3           vty3 0/33/1  SESSION_OPEN   cisco   3333::50        v2

Outgoing sessions
1               0/33/1  SESSION_OPEN    cisco   172.19.72.182   v2
2               0/33/1  SESSION_OPEN    cisco   3333::50        v2
```

```
RP/0/RP0/CPU0:router# clear ssh 0
```

The following output is applicable for the **clear ssh** command starting release 6.0 and later.

```
RP/0/RP0/CPU0:router# show ssh
SSH version : Cisco-2.0

id  chan pty     location        state           userid    host                 ver
authentication connection type
-----------------------------------------------------------------------------------------------
Incoming sessions
0   1    vty0    0/RP0/CPU0     SESSION_OPEN    lab       12.22.57.75          v2  rsa-pubkey
     Command-Line-Interface
0   2    vty1    0/RP0/CPU0     SESSION_OPEN    lab       12.22.57.75          v2  rsa-pubkey
     Command-Line-Interface
0   3           0/RP0/CPU0     SESSION_OPEN    cisco     12.22.57.75          v2  rsa-pubkey
     Sftp-Subsystem
1        vty7    0/RP0/CPU0     SESSION_OPEN    cisco     12.22.22.57            v1  password
       Command-Line-Interface
3   1            0/RP0/CPU0     SESSION_OPEN    lab       12.22.57.75            v2  password
       Netconf-Subsystem
4   1    vty3    0/RP0/CPU0     SESSION_OPEN    lab       192.168.1.55           v2  password
       Command-Line-Interface

Outgoing sessions
1                0/RP0/CPU0     SESSION_OPEN    lab       192.168.1.51           v2  password

RP/0/RP0/CPU0:router# clear ssh 0
```

clear netconf-yang agent session

To clear the specified netconf agent session, use the **clear netconf-yang agent session** in EXEC mode.

clear netconf-yang agent session *session-id*

Syntax Description

session-id	The session-id which needs to be cleared.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The **show netconf-yang clients** command can be used to get the required session-id(s).

Task ID

Task ID	Operation
config-services	read, write

Example

This example shows how to use the **clear netconf-yang agent session** command:

```
RP/0/RP0/CPU0:router (config) #  clear netconf-yang agent session 32125
```

netconf-yang agent ssh

To enable netconf agent over SSH (Secure Shell) , use the **netconf-yang agent ssh** command in the global configuration mode. To disable netconf, use the **no** form of the command.

netconf-yang agent ssh
no netconf-yang agent ssh

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global Configuration

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

SSH is currently the supported transport method for Netconf.

Task ID

Task ID	Operation
config-services	read, write

Example

This example shows how to use the **netconf-yang agent ssh** command:

```
RP/0/RP0/CPU0:router (config) #  netconf-yang agent ssh
```

sftp

To start the secure FTP (SFTP) client, use the **sftp** command.

sftp [*username* **@** *host* **:** *remote-filenam* **e**] *source-filename dest-filename* [**port** *port-num*] [**source-interface** *type interface-path-id*] [**vrf** *vrf-name*]

Syntax Description

username	(Optional) Name of the user performing the file transfer. The at symbol (@) following the username is required.
hostname:remote-filename	(Optional) Name of the Secure Shell File Transfer Protocol (SFTP) server. The colon (:) following the hostname is required.
source-filename	SFTP source, including the path.
dest-filename	SFTP destination, including the path.
port *port-num*	Specifies the non-default port number of the server to which the SFTP client on the router attempts a connection. The port number ranges from 1025 - 65535.
source-interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
type	Interface type. For more information, use the question mark (**?**) online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the **show interfaces** command in XR EXEC mode to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.
vrf *vrf-name*	Specifies the name of the VRF associated with the source interface.

Command Default

If no *username* argument is provided, the login name on the router is used. If no *hostname* argument is provided, the file is considered local.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.7.1	Modified the command to include the **port** option that specifies the non-default port for outbound connections.
Release 6.0	This command was introduced.

Usage Guidelines

SFTP provides for the secure (and authenticated) copying of files between a router and a remote host. Like the **copy** command, the **sftp** command can be invoked only in XR EXEC mode.

If a username is not provided, the login name on the router is used as the default. If a host name is not provided, the file is considered local.

If the source interface is specified in the **sftp** command, the **sftp** interface takes precedence over the interface specified in the **ssh client source-interface** command.

When the file destination is a local path, all of the source files should be on remote hosts, and vice versa.

When multiple source files exist, the destination should be a preexisting directory. Otherwise, the destination can be either a directory name or destination filename. The file source cannot be a directory name.

If you download files from different remote hosts, that is, the source points to different remote hosts, the SFTP client spawns SSH instances for each host, which may result in multiple prompts for user authentication.

If you have configured a non-default SSH server port on the router, then the SCP and SFTP services also use that SSH port for their connections. The **port** option to specify the non-default port number is available for the **ssh** command also.

The non-default SSH port number is supported only for SSHv2 and only on Cisco IOS XR SSH; not on CiscoSSH, the Open-SSH-based implementation of SSH. For more details, see *Non-default SSH Port* section in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Task ID

Task ID	Operations
crypto	execute
basic-services	execute

Examples

In the following example, user *abc* is downloading the file *ssh.diff* from the SFTP server *ena-view1* to *disk0:*

```
RP/0/RP0/CPU0:router#sftp abc@ena-view1:ssh.diff disk0
```

In the following example, user *abc* is uploading multiple files from disk 0:/sam_* to /users/abc/ on a remote SFTP server called ena-view1:

```
RP/0/RP0/CPU0:router# sftp disk0:/sam_* abc@ena-view1:/users/abc/
```

In the following example, user *admin* is downloading the file *run* from *disk0a:* to *disk0:/v6copy* on a local SFTP server using an IPv6 address:

```
RP/0/RP0/CPU0:router#sftp admin@[2:2:2::2]:disk0a:/run disk0:/V6copy
Connecting to 2:2:2::2...
Password:

disk0a:/run
  Transferred 308413 Bytes
  308413 bytes copied in 0 sec (338172)bytes/sec

RP/0/RP0/CPU0:router#dir disk0:/V6copy

Directory of disk0:
```

```
70144       -rwx  308413      Sun Oct 16 23:06:52 2011  V6copy

2102657024 bytes total (1537638400 bytes free)
```

In the following example, user *admin* is uploading the file *v6copy* from *disk0:* to *disk0a:/v6back* on a local SFTP server using an IPv6 address:

```
RP/0/RP0/CPU0:router#sftp disk0:/V6copy admin@[2:2:2::2]:disk0a:/v6back
Connecting to 2:2:2::2...
Password:

/disk0:/V6copy
  Transferred 308413 Bytes
  308413 bytes copied in 0 sec (421329)bytes/sec

RP/0/RP0/CPU0:router#dir disk0a:/v6back

Directory of disk0a:

66016       -rwx  308413      Sun Oct 16 23:07:28 2011  v6back

2102788096 bytes total (2098987008 bytes free)
```

In the following example, user *admin* is downloading the file *sampfile* from *disk0:* to *disk0a:/sampfile_v4* on a local SFTP server using an IPv4 address:

```
RP/0/RP0/CPU0:router#sftp admin@2.2.2.2:disk0:/sampfile disk0a:/sampfile_v4
Connecting to 2.2.2.2...
Password:

disk0:/sampfile
  Transferred 986 Bytes
  986 bytes copied in 0 sec (493000)bytes/sec

RP/0/RP0/CPU0:router#dir disk0a:/sampfile_v4

Directory of disk0a:

131520      -rwx  986         Tue Oct 18 05:37:00 2011  sampfile_v4

502710272 bytes total (502001664 bytes free)
```

In the following example, user *admin* is uploading the file *sampfile_v4* from *disk0a:* to *disk0:/sampfile_back* on a local SFTP server using an IPv4 address:

```
RP/0/RP0/CPU0:router#sftp disk0a:/sampfile_v4 admin@2.2.2.2:disk0:/sampfile_back
Connecting to 2.2.2.2...
Password:

disk0a:/sampfile_v4
  Transferred 986 Bytes
  986 bytes copied in 0 sec (564000)bytes/sec

RP/0/RP0/CPU0:router#dir disk0:/sampfile_back

Directory of disk0:

121765      -rwx  986         Tue Oct 18 05:39:00 2011  sampfile_back
```

```
524501272 bytes total (512507614 bytes free)
```

This example shows how to connect to the non-default port of a remote SFTP server and download a file to the local *disk0:* on the router.

```
RP/0/RP0/CPU0:router#sftp user1@198.51.100.1:disk0:/test-file port 5525 disk0
```

sftp (Interactive Mode)

To enable users to start the secure FTP (SFTP) client, use the **sftp** command.

sftp [*username* **@** *host* **:** *remote-filenam* **e**] [**port** *port-num*] [**source-interface** *type interface-path-id*]

Syntax Description

username	(Optional) Name of the user performing the file transfer. The at symbol (@) following the username is required.
hostname:remote-filename	(Optional) Name of the Secure Shell File Transfer Protocol (SFTP) server. The colon (:) following the hostname is required.
port *port-num*	Specifies the non-default port number of the server to which the SFTP client on the router attempts a connection. The port number ranges from 1025 - 65535.
source-interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
type	Interface type. For more information, use the question mark (**?**) online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the **show interfaces** command in XR EXEC mode to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.

Command Default

If no *username* argument is provided, the login name on the router is used. If no *hostname* argument is provided, the file is considered local.

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.7.1	Modified the command to include the **port** option that specifies the non-default port for outbound connections.
Release 6.0	This command was introduced.

Usage Guidelines

The SFTP client, in the interactive mode, creates a secure SSH channel where the user can enter any supported command. When a user starts the SFTP client in an interactive mode, the SFTP client process creates a secure SSH channel and opens an editor where user can enter any supported command.

More than one request can be sent to the SFTP server to execute the commands. While there is no limit on the number of 'non-acknowledged' or outstanding requests to the server, the server might buffer or queue these requests for convenience. Therefore, there might be a logical sequence to the order of requests.

The following unix based commands are supported in the interactive mode:

- bye
- **cd** <*path*>
- **chmod** <*mode*> <*path*>
- exit
- **get** <*remote-path*> *[local-path]*
- help
- **ls** *[-alt] [path]*
- **mkdir <path>**
- **put** <*local-path*> *[remote-path]*
- pwd
- quit
- **rename <old-path> <new-path>**
- **rmdir <path>**
- **rm <path>**

The following commands are not supported:

- lcd, lls, lpwd, lumask, lmkdir
- ln, symlink
- chgrp, chown
- !, !command
- ?
- mget, mput

If you have configured a non-default SSH server port on the router, then the SCP and SFTP services also use that SSH port for their connections. The **port** option to specify the non-default port number is available for the **ssh** command also.

The non-default SSH port number is supported only for SSHv2 and only on Cisco IOS XR SSH; not on CiscoSSH, the Open-SSH-based implementation of SSH. For more details, see *Non-default SSH Port* section in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

Task ID

Task ID	Operations
crypto	execute
basic-services	execute

Examples

In the following example, user *admin* is downloading and uploading a file from/to an external SFTP server using an IPv6 address:

```
RP/0/RP0/CPU0:router#sftp admin@[2:2:2::2]

Connecting to 2:2:2::2...
Password:
sftp> pwd
Remote working directory: /
sftp> cd /auto/tftp-server1-users5/admin
sftp> get frmRouter /disk0:/frmRouterdownoad

/auto/tftp-server1-users5/admin/frmRouter
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (27684)bytes/sec
sftp> put /disk0:/frmRouterdownoad againtoServer

/disk0:/frmRouterdownoad
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (14747)bytes/sec
sftp>
```

In the following example, user *abc* is downloading and uploading a file from/to an external SFTP server using an IPv4 address:

```
RP/0/RP0/CPU0:router#sftp abc@2.2.2.2
Connecting to 2.2.2.2...
Password:
sftp> pwd
Remote working directory: /
sftp> cd /auto/tftp-server1-users5/abc
sftp> get frmRouter /disk0:/frmRouterdownoad

/auto/tftp-server1-users5/abc/frmRouter
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (27684)bytes/sec
sftp> put /disk0:/frmRouterdownoad againtoServer

/disk0:/frmRouterdownoad
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (14747)bytes/sec
sftp>
```

show netconf-yang clients

To display the client details for netconf-yang, use the **show netconf-yang clients** command in XR EXEC mode.

show netconf-yang clients

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show netconf-yang clients** command:

```
RP/0/RP0/CPU0:router (config) #  sh netconf-yang clients
Netconf clients
client session ID|   NC version|    client connect time|        last OP time|        last
OP type|    <lock>|
 22969|                    1.1|          0d  0h  0m  2s|            11:11:24|
close-session|        No|
 15389|                    1.1|          0d  0h  0m  1s|            11:11:25|
get-config|        No|
```

Table 9: Field descriptions

Field name	Description
Client session ID	Assigned session identifier
NC version	Version of the Netconf client as advertised in the hello message
Client connection time	Time elapsed since the client was connected
Last OP time	Last operation time
Last OP type	Last operation type

Lock (yes or no)	To check if the session holds a lock on the configuration datastore

show netconf-yang statistics

To display the statistical details for netconf-yang, use the **show netconf-yang statistics** command in System Admin EXEC mode.

show netconf-yang statistics

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show netconf-yang statistics** command:

```
RP/0/RP0/CPU0:router (config) #  sh netconf-yang statistics
Summary statistics
                        # requests|             total time|   min time per request|   max
 time per request|   avg time per request|
other                             0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
close-session                     4|       0h  0m  0s   3ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
kill-session                      0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get-schema                        0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get                               0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get-config                        1|       0h  0m  0s   1ms|       0h  0m  0s   1ms|
 0h  0m  0s   1ms|       0h  0m  0s   1ms|
edit-config                       3|       0h  0m  0s   2ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
commit                            0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
cancel-commit                     0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
lock                              0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
unlock                            0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
```

```
discard-changes                     0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
validate                            0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
xml parse                           8|       0h  0m  0s   4ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
netconf processor                   8|       0h  0m  0s   6ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
```

Table 10: Field descriptions

Field name	Description
Requests	Total number of processed requests of a given type
Total time	Total processing time of all requests of a given type
Min time per request	Minimum processing time for a request of a given type
Max time per request	Maximum processing time for a request of a given type
Avg time per request	Average processing time for a request type

show ssh

To display all incoming and outgoing connections to the router, use the **show ssh** command.

show ssh

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show ssh** command to display all incoming and outgoing Secure Shell (SSH) Version 1 (SSHv1) and SSH Version 2 (SSHv2) connections.

The connection type field in the command output of **show ssh** command shows as **port-forwarded local** for SSH port-forwarded sessions.

Use the **show ssh server** command to see the details of the SSH server. The **Port Forwarding** column shows as **local** for the port-forwarded session. Whereas, for a regular SSH session, the field displays as **disabled**.

Task ID

Task ID	Operations
crypto	read

Examples

This is sample output from the **show ssh** command when SSH is enabled:

```
RP/0/RP0/CPU0:router# show ssh

SSH version : Cisco-2.0

id  pty   location    state         userid    host          ver       authentication
-----------------------------------------------------------------------------------------------------
Incoming sessions

Outgoing sessions
1         0/3/CPU0    SESSION_OPEN    lab     12.22.57.     v2        password
2         0/3/CPU0    SESSION_OPEN    lab     12.22.57.75   v2        keyboard-interactive
```

The following output is applicable for the **show ssh** command starting IOS-XR 6.0 releases and later.

```
RP/0/RP0/CPU0:router# show ssh
SSH version : Cisco-2.0
```

```
id  chan pty     location        state           userid    host                 ver
authentication connection type
-----------------------------------------------------------------------------------------------
Incoming sessions
0   1    vty0    0/RP0/CPU0     SESSION_OPEN    lab       12.22.57.75          v2  rsa-pubkey
     Command-Line-Interface
0   2    vty1    0/RP0/CPU0     SESSION_OPEN    lab       12.22.57.75          v2  rsa-pubkey
     Command-Line-Interface
0   3           0/RP0/CPU0     SESSION_OPEN    cisco     12.22.57.75          v2  rsa-pubkey
     Sftp-Subsystem
1        vty7    0/RP0/CPU0     SESSION_OPEN    cisco     12.22.22.57            v1  password
       Command-Line-Interface
3   1           0/RP0/CPU0     SESSION_OPEN    lab       12.22.57.75            v2  password
       Netconf-Subsystem
4   1    vty3    0/RP0/CPU0     SESSION_OPEN    lab       192.168.1.55           v2  password
       Command-Line-Interface

Outgoing sessions
1               0/RP0/CPU0     SESSION_OPEN    lab       192.168.1.51           v2  password
```

This table describes significant fields shown in the display.

Table 11: show ssh Field Descriptions

Field	Description
session	Session identifier for the incoming and outgoing SSH connections.
chan	Channel identifier for incoming (v2) SSH connections. NULL for SSH v1 sessions.
pty	pty-id allocated for the incoming session. Null for outgoing SSH connection.
location	Specifies the location of the SSH server for an incoming connection. For an outgoing connection, location specifies from which route processor the SSH session is initiated.
state	The SSH state that the connection is currently in.
userid	Authentication, authorization and accounting (AAA) username used to connect to or from the router.
host	IP address of the remote peer.
ver	Specifies if the connection type is SSHv1 or SSHv2.
authentication	Specifies the type of authentication method chosen by the user.
connection type	Specifies which application is performed over this connection (Command-Line-Interface, Remote-Command, Scp, Sftp-Subsystem, or Netconf-Subsystem)

The following is a sample output of SSH port-forwarded session:

```
Router#show ssh

Wed Oct 14 11:22:05.575 UTC
SSH version : Cisco-2.0
```

```
id chan pty location   state        userid host          ver authentication connection type
-----------------------------------------------------------------------------------------------
Incoming sessions
15 1    XXX 0/RP0/CPU0 SESSION_OPEN admin  192.168.122.1 v2  password
port-forwarded-local

Outgoing sessions

Router#
```

The following is a sample output of **show ssh server** command with SSH port forwarding enabled:

```
Router#show ssh server
Tue Sep  7 17:43:22.483 IST
---------------------
SSH Server Parameters
---------------------

Current supported versions := v2
                  SSH port := 22
                  SSH vrfs := vrfname:=default(v4-acl:=, v6-acl:=)
              Netconf Port := 830
              Netconf Vrfs := vrfname:=default(v4-acl:=, v6-acl:=)

 Algorithms
---------------
        Hostkey Algorithms :=
x509v3-ssh-rsa,ecdsa-sha2-nistp521,ecdsa-sha2-nistp384,ecdsa-sha2-nistp256,rsa-sha2-512,rsa-sha2-256,ssh-rsa,ssh-dsa,ssh-ed25519

   Key-Exchange Algorithms :=
ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-hellman-group14-sha1
     Encryption Algorithms :=
aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com
            Mac Algorithms := hmac-sha2-512,hmac-sha2-256,hmac-sha1

 Authentication Method Supported
------------------------------------
                 PublicKey := Yes
                  Password := Yes
      Keyboard-Interactive := Yes
         Certificate Based := Yes

 Others
------------
                     DSCP  := 0
                Ratelimit  := 600
             Sessionlimit  := 110
                Rekeytime  := 30
       Server rekeyvolume  := 1024
  TCP window scale factor  := 1
            Backup Server  := Disabled
          Host Trustpoint  :=
          User Trustpoint  := tes,test,x509user
          Port Forwarding  := local
Max Authentication Limit  := 16
     Certificate username  := Common name(CN) User principle name(UPN)
Router#
```

show ssh history

To display the last hundred SSH connections that were terminated, use the **show ssh history** command in XR EXEC mode.

show ssh history

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.4.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
crypto	read

Examples

The following is sample output from the **show ssh history** command to display the last hundred SSH sessions that were teminated:

```
RP/0/RP0/CPU0:router# show ssh history

SSH version : Cisco-2.0

id       chan pty     location        userid    host                  ver authentication
connection type
-----------------------------------------------------------------------------------------------------
Incoming sessions
1        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
2        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
3        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
4        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
5        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
6        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
7        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
8        1    XXXXX   0/RP0/CPU0      root      10.105.227.252        v2  password
Netconf-Subsystem
```

```
9        1     vty0     0/RP0/CPU0       root       10.196.98.106          v2  key-intr
Command-Line-Interface
```

Pty – VTY number used. This is represented as ‘XXXX’ when connection type is SFTP, SCP or Netconf.

show ssh history details

To display the last hundred SSH connections that were terminated, and also the start and end time of the session, use the **show ssh history details** command in XR EXEC mode.

show ssh history details

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.4.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
crypto	read

Examples

The following is sample output from the **show ssh history details** command to display the last hundred SSH sessions that were teminated along with the start and end time of the sessions:

```
RP/0/RP0/CPU0:router# show ssh history details

SSH version : Cisco-2.0

id      key-exchange           pubkey               incipher    outcipher   inmac
outmac          start_time              end_time
------------------------------------------------------------------------------------------

Incoming Session
1       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   14-02-18 14:00:39       14-02-18 14:00:41
2       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   14-02-18 16:21:54       14-02-18 16:21:55
3       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   14-02-18 16:22:18       14-02-18 16:22:19
4       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   15-02-18 12:17:44       15-02-18 12:17:46
5       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   15-02-18 12:18:16       15-02-18 12:18:17
6       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   15-02-18 14:44:08       15-02-18 14:44:09
7       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   15-02-18 14:50:15       15-02-18 14:50:16
8       ecdh-sha2-nistp256     ssh-rsa              aes128-ctr  aes128-ctr  hmac-sha2-256
```

```
hmac-sha2-256   15-02-18 14:50:52       15-02-18 14:50:53
9       ecdh-sha2-nistp256     ssh-rsa             aes128-ctr  aes128-ctr  hmac-sha2-256
hmac-sha2-256   15-02-18 15:31:26       15-02-18 15:31:38
```

This table describes the significant fields shown in the display.

Table 12: Field Descriptions

Field	Description
session	Session identifier for the incoming and outgoing SSH connections.
key-exchange	Key exchange algorithm chosen by both peers to authenticate each other.
pubkey	Public key algorithm chosen for key exchange.
incipher	Encryption cipher chosen for the receiver traffic.
outcipher	Encryption cipher chosen for the transmitter traffic.
inmac	Authentication (message digest) algorithm chosen for the receiver traffic.
outmac	Authentication (message digest) algorithm chosen for the transmitter traffic.
start_time	Start time of the session.
end_time	End time of the session.

show ssh session details

To display the details for all incoming and outgoing Secure Shell Version 2 (SSHv2) connections, use the **show ssh session details** command.

show ssh session details

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **show ssh session details** command to display a detailed report of the SSHv2 connections to or from the router, including the cipher chosen for the specific session.

Task ID

Task ID	Operations
crypto	read

Examples

The following is sample output from the **show ssh session details** command to display the details for all the incoming and outgoing SSHv2 connections:

```
RP/0/RP0/CPU0:router# show ssh session details

SSH version: Cisco-2.0
session     key-exchange   pubkey   incipher   outcipher    inmac     outmac
--------------------------------------------------------------------------------
Incoming Session

0           diffie-hellman ssh-dss  3des-cbc   3des-cbc     hmac-md5  hmac-md5

Outgoing connection

1           diffie-hellman ssh-dss  3des-cbc   3des-cbc     hmac-md5  hmac-md5
```

This table describes the significant fields shown in the display.

Table 13: show ssh session details Field Descriptions

Field	Description
session	Session identifier for the incoming and outgoing SSH connections.
key-exchange	Key exchange algorithm chosen by both peers to authenticate each other.

Field	Description
pubkey	Public key algorithm chosen for key exchange.
incipher	Encryption cipher chosen for the Rx traffic.
outcipher	Encryption cipher chosen for the Tx traffic.
inmac	Authentication (message digest) algorithm chosen for the Rx traffic.
outmac	Authentication (message digest) algorithm chosen for the Tx traffic.

show tech-support ssh

To automatically run show commands that display system information, use the show tech-support command, use the **show tech-support ssh** command in XR EXEC mode.

show tech-support ssh

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR EXEC mode

Command History

Release	Modification
Release 6.4.1	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
crypto	read

Examples The following is sample output from the **show tech-support ssh** command:

```
RP/0/RP0/CPU0:router# show tech-support ssh
++ Show tech start time: 2018-Feb-20.123016.IST ++
Tue Feb 20 12:30:27 IST 2018 Waiting for gathering to complete
.............................
Tue Feb 20 12:32:35 IST 2018 Compressing show tech output
Show tech output available at 0/RP0/CPU0 :
/harddisk:/showtech/showtech-ssh-2018-Feb-20.123016.IST.tgz
++ Show tech end time: 2018-Feb-20.123236.IST ++
RP/0/RP0/CPU0:turin-sec1#
```

The **show tech-support ssh** command collects the output of these CLI:

Command	Description
show logging	Displays the contents of the logging buffer.
show context location all	
show running-config	Displays the contents of the currently running configuration or a subset of that configuration.
show ip int brief	Displays brief information about each interface.

Command	Description
show ssh	Displays all incoming and outgoing connections to the router.
show ssh session details	Displays the details for all the incoming and outgoing SSHv2 connections, to the router.
show ssh rekey	Displays session rekey details such as session id, session rekey count, time to rekey, data to rekey.
show ssh history	Displays the last hundred SSH connections that were terminated.
show tty trace info all all	
show tty trace error all all	

ssh

To start the Secure Shell (SSH) client connection and enable an outbound connection to an SSH server, use the **ssh** command.

ssh { *ipv4-address* [**port** *port-num*] | *ipv6-address* [**port** *port-num*] | *hostname* [**port** *port-num*] } [**username** *user-id*] [**cipher aes** { **128-cbc** | **192-cbc** | **256-cbc** }] [**source-interface** *type interface-path-id*] [**command** *command-name*]

Syntax Description

ipv4-address	IPv4 address in A:B:C:D format.
ipv6-address	IPv6 address in X:X::X format.
hostname	Hostname of the remote node. If the hostname has both IPv4 and IPv6 addresses, the IPv6 address is used.
port *port-num*	Specifies the non-default SSH port number of the remote SSH server to which the SSH client on the router attempts a connection. The port number ranges from 1025 - 65535.
username *user-id*	(Optional) Specifies the username to use when logging in on the remote networking device running the SSH server. If no user ID is specified, the default is the current user ID.
cipheraes	(Optional) Specifies Advanced Encryption Standard (AES) as the cipher for the SSH client connection. **Note** If there is no specification of a particular cipher by the administrator, the client proposes 3DES as the default to ensure compatibility.
128-CBC	128-bit keys in CBC mode.
192-CBC	192-bit keys in CBC mode.
256-CBC	256-bit keys in CBC mode.
source interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
type	Interface type. For more information, use the question mark (**?**)online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the**showinterfaces** command in XR EXEC mode to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark(**?**)online help function.
command	(Optional) Specifies a remote command. Adding this keyword prompts the SSHv2 server to parse and execute the**ssh**command in non-interactive mode instead of initiating the interactive session.

Command Default

3DES cipher

Command Modes

XR EXEC mode

Command History

Release	Modification
Release 7.7.1	Modified the command to include the **port** option that specifies the non-default port for outbound SSH connections.
Release 6.0	This command was introduced.

Usage Guidelines

Use the **ssh** command to make an outbound client connection. The SSH client tries to make an SSHv2 connection to the remote peer. If the remote peer supports only the SSHv1 server, it internally spawns an SSHv1 connection to the remote server. The process of the remote peer version detection and spawning the appropriate client connection is transparent to the user.

If a VRF is specified in the **ssh** command, the **ssh** interface takes precedence over the interface specified in the ssh client source-interface, on page 177 command.

When you configure the **cipher aes** keyword, an SSH client makes a proposal, including one or more of the key sizes you specified, as part of its request to the SSH server. The SSH server chooses the best possible cipher, based both on which ciphers that server supports and on the client proposal.

Note AES encryption algorithm is not supported on the SSHv1 server and client. Any requests for an AES cipher sent by an SSHv2 client to an SSHv1 server are ignored, with the server using 3DES instead.

A VRF is required to run SSH, although this may be either the default VRF or a VRF specified by the user. If no VRF is specified while configuring the ssh client source-interface, on page 177 or ssh client knownhost, on page 176 commands, the default VRF is assumed.

Use the **command** keyword to enable the SSHv2 server to parse and execute the **ssh** command in non-interactive mode instead of initiating an interactive session.

The non-default SSH port number is supported only for SSHv2 and only on Cisco IOS XR SSH; not on CiscoSSH, the Open-SSH-based implementation of SSH. For more details, see *Non-default SSH Port* section in the *System Security Configuration Guide for Cisco NCS 5000 Series Routers*.

If you have configured a non-default SSH server port on the router, then the SCP and SFTP services also use that SSH port for their connections. The **port** option to specify the non-default port number is available for the **scp** and **sftp** commands also.

Among the NCS540 router variants, the non-default **port** option is applicable only for the following variants:

- N540-ACC-SYS
- N540X-ACC-SYS
- N540-24Z8Q2C-SYS

Task ID

Task ID	Operations
crypto	execute

Task ID	Operations
basic-services	execute

Examples

The following sample output is from the **ssh** command to enable an outbound SSH client connection:

```
Router# ssh vrf green  username userabc

Password:
Remote-host>
```

ssh algorithms cipher

To configure the list of supported SSH algorithms on the client or on the server, use the **ssh client algorithms cipher** command or **ssh server algorithms cipher** command in XR Config mode. To remove the configuration, use the **no** form of this command.

ssh {**client** | **server**} **algorithms cipher** {**aes256-cbc** | **aes256-ctr** | **aes192-ctr** | **aes192-cbc** | **aes128-ctr** | **aes128-cbc** | **aes128-gcm@openssh.com** | **aes256-gcm@openssh.com** | **3des-cbc**}

Syntax Description

client	Configures the list of supported SSH algorithms on the client.
server	Configures the list of supported SSH algorithms on the server.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
crypto	read, write

This example shows how to enable CTR cipher on the client and CBC cipher on the server:

```
Router1#ssh client algorithms cipher aes128-ctr aes192-ctr aes256-ctr

Router1#ssh server algorithms cipher aes128-cbc aes192-cbc aes256-cbc 3des-cbc
```

Related Commands

Command	Description
ssh client enable cipher , on page 174	Enables CBC mode ciphers on the SSH client.
ssh server enable cipher, on page 182	Enables CBC mode ciphers on the SSH server.

ssh client auth-method

To set the preferred order of SSH client authentication methods to be negotiated with the SSH server while establishing SSH sessions, use the **ssh client auth-method** command in the XR Config mode. To revert to the default order of SSH client authentication methods, use the **no** form of this command.

```
ssh client auth-method list-of-auth-method
```

Syntax Description

list-of-auth-method	Specifies the list of SSH client authentication methods in the respective order. The available options are: • **keyboard-interactive** • **password** • **public-key**

Command Default

None

Command Modes

XR Config

Command History

Release	Modification
Release 7.9.2/Release 7.10.1	This command was introduced.

Usage Guidelines

The default order of SSH client authentication methods on Cisco IOS XR routers is as follows:

- On routers running Cisco IOS XR SSH:
  - **public-key**, **password** and **keyboard-interactive**
- On routers running CiscoSSH (open source-based SSH):
  - **public-key**, **keyboard-interactive** and **password**

Task ID

Task ID	Operation
crypto	read, write

This example shows how to set the order of SSH client authentication methods in such a way that public key authentication is negotiated first, followed by keyboard-interactive, and then password-based authentication.

```
Router#configure
Router(config)#ssh client auth-method public-key keyboard-interactive password
Router(config-ssh)#commit
```

ssh client enable cipher

To enable the CBC mode ciphers 3DES-CBC and/or AES-CBC for an SSH client connection, use the **ssh client enable cipher** command in XR Config mode. To disable the ciphers, use the **no** form of this command.

ssh client enable cipher {**aes-cbc** | **3des-cbc**}

Syntax Description

3des-cbc	Specifies that the 3DES-CBC cipher be enabled for the SSH client connection.
aes-cbc	Specifies that the AES-CBC cipher be enabled for the SSH client connection.

Command Default

CBC mode ciphers are disabled.

Command Modes

Global Configuration

Command History

Release	Modification
Release 6.3.1	This command was introduced.

Usage Guidelines

The support for CBC ciphers were disabled by default, from Cisco IOS XR Software Release 6.1.2. Hence, **ssh client enable cipher** and **ssh server enable cipher** commands were introduced to explicitly enable CBC ciphers in required scenarios.

If a client tries to reach the router which acts as a server with CBC cipher, and if the CBC cipher is not explicitly enabled on that router, then the system displays an error message:

```
ssh root@x.x.x. -c aes128-cbc
Unable to negotiate with x.x.x.x port 22: no matching cipher found.
Their offer: aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com
```

You must configure **ssh server enable cipher aes-cbc** command in this case, to connect to the router using the CBC cipher.

Task ID

Task ID	Operation
crypto	read, write

Examples

The following example shows how to enable the 3DES-CBC and AES-CBC ciphers for an SSH client connection:

```
Router# configure
```

```
Router(config)# ssh client enable cipher aes-cbc 3des-cbc
Router(config)# commit
```

Related Commands

Command	Description
ssh algorithms cipher, on page 172	Configures the list of supported SSH algorithms on the client or on the server.
ssh server enable cipher, on page 182	Enables CBC mode ciphers on the SSH server.

ssh client knownhost

To authenticate a server public key (pubkey), use the **ssh client knownhost** command. To disable authentication of a server pubkey, use the **no** form of this command.

ssh client knownhost device : /filename
no ssh client knownhost device : /filename

Syntax Description

device:/*filename*	Complete path of the filename (for example, slot0:/server_pubkey). The colon (:) and slash (/) are required.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The *server pubkey* is a cryptographic system that uses two keys at the client end—a public key known to everyone and a private, or secret, key known only to the owner of the keys. In the absence of certificates, the server pubkey is transported to the client through an out-of-band secure channel. The client stores this pubkey in its local database and compares this key against the key supplied by the server during the early stage of key negotiation for a session-building handshake. If the key is not matched or no key is found in the local database of the client, users are prompted to either accept or reject the session.

The operative assumption is that the first time the server pubkey is retrieved through an out-of-band secure channel, it is stored in the local database. This process is identical to the current model adapted by Secure Shell (SSH) implementations in the UNIX environment.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following sample output is from the **ssh client knownhost** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh client knownhost disk0:/ssh.knownhost
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router# ssh host1 username user1234
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? yes
Password:
RP/0/RP0/CPU0:host1# exit
RP/0/RP0/CPU0:router# ssh host1 username user1234
```

ssh client source-interface

To specify the source IP address of a selected interface for all outgoing Secure Shell (SSH) connections, use the **ssh client source-interface** command. To disable use of the specified interface IP address, use the **no** form of this command.

ssh client source-interface *type interface-path-id*
no ssh client source-interface *type interface-path-id*

Syntax Description

type	Interface type. For more information, use the question mark (?) online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (**?**) online help function.

Command Default

No source interface is used.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **ssh client source-interface** command to set the IP address of the specified interface for all outgoing SSH connections. If this command is not configured, TCP chooses the source IP address when the socket is connected, based on the outgoing interface used—which in turn is based on the route required to reach the server. This command applies to outbound shell over SSH as well as Secure Shell File Transfer Protocol (SFTP) sessions, which use the ssh client as a transport.

The source-interface configuration affects connections only to the remote host in the same address family. The system database (Sysdb) verifies that the interface specified in the command has a corresponding IP address (in the same family) configured.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following example shows how to set the IP address of the Management Ethernet interface for all outgoing SSH connections:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh client source-interface MgmtEth 0/RP0/CPU0/0
```

ssh server

To bring up the Secure Shell (SSH) server, use the **ssh server** command. To stop the SSH server, use the **no** form of this command.

ssh server
no ssh server

This command has no keywords or arguments.

Command Default

The default SSH server version is 2 (SSHv2), which falls back to 1 (SSHv1) if the incoming SSH client connection is set to SSHv1.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

The SSH server listens for an incoming client connection on port 22. This server handles both Secure Shell Version 1 (SSHv1) and SSHv2 incoming client connections for both IPv4 and IPv6 address families. To accept only Secure Shell Version 2 connections, use the ssh server v2, on page 188 command.

To verify that the SSH server is up and running, use the **show process sshd** command.

Task ID

Task ID	Operations
crypto	read, write

Examples

In the following example, how to bring up the the SSH server:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server
```

ssh server algorithms host-key

To configure the allowed SSH host-key pair algorithms from the list of auto-generated host-key pairs on the SSH server, use the **ssh server algorithms host-key** command in XR Config mode. To remove the configuration, use the **no** form of this command.

ssh server algorithms host-key { **dsa** | **ecdsa-nistp256** | **ecdsa-nistp384** | **ecdsa-nistp521** | **rsa** }

Syntax Description

• **dsa** • **ecdsa-nistp256** • **ecdsa-nistp384** • **ecdsa-nistp521** • **rsa**	Selects the specified host keys to be offered to the SSH client. While configuring this, you can specify the algorithms in any order.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.1	This command was introduced.

Usage Guidelines

This configuration is optional. If this configuration is not present, it is assumed that all the SSH host-key pairs are configured. In that case, the SSH client is allowed to connect to the SSH sever with any of the host-key pairs.

You can also use the **crypto key zeroize** command to remove the SSH algorithms that are not required.

With the introduction of the automatic generation of SSH host-key pairs, the **show crypto key mypubkey** command output displays key information of all the keys that are auto-generated. Before its introduction, the output of this command displayed key information of only those host-key pairs that were explicitly configured using the **crypto key generate** command.

Task ID

Task ID	Operation
crypto	read, write

This example shows how to select the **ecdsa** algorithm from the list of auto-generated host-key pairs on the SSH server:

```
Router#ssh server algorithms host-key ecdsa-nistp521
```

ssh server disable hmac

To disable HMAC cryptographic algorithm on the SSH server, use the **ssh server disable hmac** command, and to disable HMAC cryptographic algorithm on the SSH client, use the **ssh client disable hmac** command in XR Config mode. To disable this feature, use the **no** form of this command.

ssh {**client** | **server**} **disable hmac** {**hmac-sha1** | **hmac-sha2-512**}

Syntax Description

hmac-sha1	Disables the SHA-1 HMAC cryptographic algorithm.
hmac-sha2-512	Disables the SHA-2 HMAC cryptographic algorithm. **Note** This option is available only for the **server**.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.0.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
crypto	read, write

This example shows how to disable SHA1 HMAC cryptographic algorithm on the SSH client:

```
Router#ssh client disable hmac hmac-sha1
```

This example shows how to disable SHA-2 HMAC cryptographic algorithm on the SSH server:

```
Router#ssh server disable hmac hmac-sha2-512
```

ssh server enable cipher

To enable CBC mode ciphers 3DES-CBC and/or AES-CBC for an SSH server connection, use the **ssh server enable cipher** command in XR Config mode. To disable the ciphers, use the **no** form of this command.

ssh server enable cipher {**aes-cbc** | **3des-cbc**}

Syntax Description

3des-cbc	Specifies that the 3DES-CBC cipher be enabled for the SSH server connection.
aes-cbc	Specifies that the AES-CBC cipher be enabled for the SSH server connection.

Command Default

CBC mode ciphers are disabled.

Command Modes

Global Configuration

Command History

Release	Modification
Release 6.3.1	This command was introduced.

Usage Guidelines

The support for CBC ciphers were disabled by default, from Cisco IOS XR Software Release 6.1.2. Hence, **ssh client enable cipher** and **ssh server enable cipher** commands were introduced to explicitly enable CBC ciphers in required scenarios.

Task ID

Task ID	Operation
crypto	read, write

Examples

The following example shows how to enable the 3DES-CBC and AES-CBC ciphers for an SSH server connection:

```
Router# configure
Router(config)# ssh server enable cipher aes-cbc 3des-cbc
Router(config)# commit
```

Related Commands

Command	Description
ssh algorithms cipher, on page 172	Configures the list of supported SSH algorithms on the client or on the server.
ssh client enable cipher , on page 174	Enables CBC mode ciphers on the SSH client.

ssh server logging

To enable SSH server logging, use the **ssh server logging** command. To discontinue SSH server logging, use the **no** form of this command.

ssh server logging
no ssh server logging

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Only SSHv2 client connections are allowed.

Once you configure the logging, the following messages are displayed:

- Warning: The requested term-type is not supported
- SSH v2 connection from %s succeeded (*user:%s, cipher:%s, mac:%s, pty:%s*)

The warning message appears if you try to connect using an unsupported terminal type. Routers running the Cisco IOS XR software support only the vt100 terminal type.

The second message confirms a successful login.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following example shows the initiation of an SSH server logging:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server logging
```

ssh server port

To configure a non-default port for the SSH server, use the **ssh server port** command in XR Config mode. To remove the configuration and to change the SSH port number to the default port (22), use the **no** form of this command.

ssh server port *port-number*

Syntax Description

port-number	Specifies the non-default SSH port number. The limit ranges from 5520 to 5529.

Command Default

Disabled, by default.

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.7.1	This command was introduced

Usage Guidelines

If this command is not configured, then the SSH server uses the default port number, 22, for all SSH, SCP and SFTP services.

Among the NCS540 router variants, this command is applicable only for the following variants:

- N540-ACC-SYS
- N540X-ACC-SYS
- N540-24Z8Q2C-SYS

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to configure a non-default SSH port for the SSH server on your router:

```
Router# configure
Router(config)# ssh server port 5520
Router(config)# commit
```

ssh server port-forwarding local

To enable SSH port forwarding feature on SSH server, use the **ssh server port-forwarding local** command in XR Config mode. To disable the feature, use the **no** form of this command.

ssh server port-forwarding local

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 7.3.2	This command was introduced.

Usage Guidelines

The Cisco IOS XR software supports SSH port forwarding only on SSH server; not on SSH client. Hence, to utilize this feature, the SSH client running at the end host must already have the support for SSH port forwarding or tunneling.

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to enable SSH port forwarding feature on SSH server:

```
Router#configure
Router(config)#ssh server port-forwarding local
Router(config)#commit
```

Related Commands

Command	Description
show ssh, on page 158	Displays all incoming and outgoing SSH connections on the router.

ssh server rate-limit

To limit the number of incoming Secure Shell (SSH) connection requests allowed per minute, use the **ssh server rate-limit** command. To return to the default value, use the **no** form of this command.

ssh server rate-limit *rate-limit*
no ssh server rate-limit

Syntax Description

rate-limit	Number of incoming SSH connection requests allowed per minute. Range is from 1 to 120. When setting it to 60 attempts per minute, it basically means that we can only allow 1 per second. If you set up 2 sessions at the same time from 2 different consoles, one of them will get rate limited. This is connection attempts to the ssh server, not bound per interface/username or anything like that. So value of 30 means 1 session per 2 seconds and so forth.

Command Default

rate-limit: 60 connection requests per minute

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **ssh server rate-limit** command to limit the incoming SSH connection requests to the configured rate. Any connection request beyond the rate limit is rejected by the SSH server. Changing the rate limit does not affect established SSH sessions.

If, for example, the *rate-limit* argument is set to 30, then 30 requests are allowed per minute, or more precisely, a two-second interval between connections is enforced.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following example shows how to set the limit of incoming SSH connection requests to 20 per minute:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server rate-limit 20
```

ssh server session-limit

To configure the number of allowable concurrent incoming Secure Shell (SSH) sessions, use the **ssh server session-limit** command. To return to the default value, use the **no** form of this command.

ssh server session-limit *sessions*

Syntax Description

sessions Number of incoming SSH sessions allowed across the router. The range is from 1 to 100.

Note Although CLI output option has 1024, you are recommended to configure session-limit not more than 100. High session count may cause resource exhaustion .

Command Default

sessions: 64 per router

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **ssh server session-limit** command to configure the limit of allowable concurrent incoming SSH connections. Outgoing connections are not part of the limit.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following example shows how to set the limit of incoming SSH connections to 50:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server session-limit 50
```

ssh server v2

To force the SSH server version to be only 2 (SSHv2), use the **ssh server v2** command. To bring down an SSH server for SSHv2, use the **no** form of this command.

ssh server v2
no ssh server v2

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Only SSHv2 client connections are allowed.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following example shows how to initiate the SSH server version to be only SSHv2:

```
 RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)# ssh server v2
```

ssh server vrf

To bring up the Secure Shell (SSH) server and to configure one or more VRFs for its use, use the **ssh server vrf** command. To stop the SSH server from receiving any further connections for the specified VRF, use the **no** form of this command. Optionally ACLs for IPv4 and IPv6 can be used to restrict access to the server before the port is opened.

ssh server vrf *vrf-name* [**ipv4 access-list** *access-list name*] [**ipv6 access-list** *access-list name*]
no ssh server vrf *vrf-name* [**ipv4 access-list** *access-list name*] [**ipv6 access-list** *access-list name*]

Syntax Description

vrf *vrf-name*	Specifies the name of the VRF to be used by the SSH server. The maximum VRF length is 32 characters. **Note** If no VRF is specified, the default VRF is assumed.
ipv4 access-list *access-list name*	Configures an IPv4 access-list for access restrictions to the ssh server. The maximum length of the access-list name length is 32 characters.
ipv6 access-list *access-list name*	Configures an IPv6 access-list for access restrictions to the ssh server. The maximum length of the access-list name length is 32 characters.

Command Default

The default SSH server version is 2 (SSHv2), which falls back to 1 (SSHv1) if the incoming SSH client connection is set to SSHv1.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

An SSH server must be configured at minimum for one VRF. If you delete all configured VRFs, including the default, the SSH server process stops. If you do not configure a specific VRF for the SSH client when applying other commands, such as **ssh client knownhost** or **ssh client source-interface** the default VRF is assumed.

To verify that the SSH server is up and running, use the **show process sshd** command.

Task ID

Task ID	Operations
crypto	read, write

Examples

In the following example, the SSH server is brought up to receive connections for VRF “green”:

```
RP/0/RP0/CPU0:router# configure
```

```
RP/0/RP0/CPU0:router(config)# ssh server vrf green
```

In the following example, the SSH server is brought up to receive connections for VRF “green” and a standard access list ipv4 access list named Internetfilter is configured:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server vrf green ipv4 access-list Internetfilter
```

ssh server netconf

To configure a port for the netconf SSH server, use the **ssh server netconf port** in the XR Config mode. To disable netconf for the configured port, use the **no** form of the command.

ssh server netconf [**port** *port-number*]
no ssh server netconf [**port** *port-number*]

Syntax Description

port-number	Port number for the netconf SSH server (default port number is 830).

Command Default

Default port number is 830.

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
crypto	read, write

Example

This example shows how to use the **ssh server netconf port** command:

```
RP/0/RP0/CPU0:router (config) #  ssh server netconf port 830
```

ssh timeout

To configure the timeout value for authentication, authorization, and accounting (AAA) user authentication, use the **ssh timeout** command. To set the timeout value to the default time, use the **no** form of this command.

ssh timeout *seconds*
no ssh timeout *seconds*

Syntax Description

seconds	Time period (in seconds) for user authentication. The range is from 5 to 120.

Command Default

seconds: 30

Command Modes

XR Config mode

Command History

Release	Modification
Release 6.0	This command was introduced.

Usage Guidelines

Use the **ssh timeout** command to configure the timeout value for user authentication to AAA. If the user fails to authenticate itself within the configured time to AAA, the connection is terminated. If no value is configured, the default value of 30 seconds is used.

Task ID

Task ID	Operations
crypto	read, write

Examples

In the following example, the timeout value for AAA user authentication is set to 60 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh timeout 60
```

INDEX

S