Authentication, Authorization, and Accounting Commands

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

To use commands of this module, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using any command, contact your AAA administrator for assistance.

For detailed information about AAA concepts, configuration tasks, and examples, see the Configuring AAA Services chapter in the System Security Configuration Guide for Cisco 8000 Series Routers.

- aaa accounting, on page 3
- aaa accounting system default, on page 5
- aaa accounting system rp-failover, on page 7
- aaa accounting update, on page 8
- aaa authentication, on page 9
- aaa authorization, on page 11
- aaa default-taskgroup, on page 14
- aaa group server radius, on page 15
- aaa group server tacacs+, on page 17
- aaa password-policy, on page 18
- accounting (line), on page 21
- authorization (line), on page 22
- deadtime (server-group configuration), on page 23
- description (AAA), on page 24
- group (AAA), on page 25
- inherit taskgroup, on page 27
- inherit usergroup, on page 28
- key (RADIUS), on page 29
- key (TACACS+), on page 31
- login authentication, on page 32
- password (AAA), on page 34
- radius-server dead-criteria time, on page 36
- radius-server dead-criteria tries, on page 38
- radius-server deadtime, on page 40
• radius-server host, on page 41
• radius-server key, on page 43
• radius-server retransmit, on page 44
• radius-server timeout, on page 45
• retransmit (RADIUS), on page 46
• secret, on page 47
• server (RADIUS), on page 49
• server (TACACS+), on page 51
• server-private (RADIUS), on page 52
• server-private (TACACS+), on page 54
• show aaa, on page 56
• show aaa password-policy, on page 62
• show radius accounting, on page 64
• show radius authentication, on page 66
• show radius, on page 68
• show radius dead-criteria, on page 70
• show radius server-groups, on page 72
• show tacacs, on page 74
• show tacacs server-groups, on page 76
• show user, on page 77
• single-connection, on page 81
• tacacs-server host, on page 82
• tacacs-server ipv4, on page 84
• tacacs-server key, on page 86
• tacacs-server timeout, on page 87
• tacacs source-interface, on page 88
• task, on page 90
• taskgroup, on page 91
• timeout login response, on page 93
• timeout (RADIUS), on page 94
• timeout (TACACS+), on page 95
• usergroup, on page 96
• username, on page 97
• users group, on page 101
• vrf (RADIUS), on page 103
• vrf (TACACS+), on page 104
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 | subscriber | 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).
- **subscriber**: Sets accounting lists for subscribers.
- **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

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

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} {broadcast | none} method

start-stop          Sends a “start accounting” notice during system bootup and a “stop accounting” notice during system shutdown or reload.

broadcast          Sets the broadcast 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 Config mode

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

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.
AAA accounting system default

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

To create an accounting list to send rp-failover or rp-switchover start or stop accounting messages, use the `aaa accounting system rp-failover` command in XR Config mode. To disable the system accounting for rp-failover, use the `no` form of this command.

```
aaa accounting system rp-failover {list_name {start-stop | stop-only} | default {start-stop | stop-only}}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>list_name</code></td>
<td>Specifies the accounting list name.</td>
</tr>
<tr>
<td><code>default</code></td>
<td>Specifies the default accounting list.</td>
</tr>
<tr>
<td><code>start-stop</code></td>
<td>Enables the start and stop records.</td>
</tr>
<tr>
<td><code>stop-only</code></td>
<td>Enables the stop records only.</td>
</tr>
</tbody>
</table>

**Command Default**

None

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This is an example of configuring the `aaa accounting system rp-failover` command for default accounting list:

```
RP/0/RP0/CPU0:router(config)# aaa accounting system rp-failover default start-stop none
```
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** *(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**
  - Release 7.0.12
  - 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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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

To create a method list for authentication, use the aaa authentication command. To disable this authentication method, use the no form of this command.

```
aaa authentication {login | ppp} {default|list-name | remote} method-list
no aaa authentication {login | onepk | ppp} {default|list-name} method-list
```

**Syntax Description**

- `login` Sets authentication lists for login.
- `onepk` Sets authentication lists for OnePk.
- `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.

**Command Default**

Default behavior applies the local authentication on all ports.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.

### Examples

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

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

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>aaa accounting</code>, on page 3</td>
<td>Creates a method list for accounting.</td>
</tr>
<tr>
<td><code>aaa authorization</code>, on page 11</td>
<td>Creates a method list for authorization.</td>
</tr>
<tr>
<td><code>aaa group server radius</code>, on page 15</td>
<td>Groups different RADIUS server hosts into distinct lists and distinct methods.</td>
</tr>
<tr>
<td><code>aaa group server tacacs+</code>, on page 17</td>
<td>Groups different TACACS+ server hosts into distinct lists and distinct methods.</td>
</tr>
<tr>
<td><code>login authentication</code>, on page 32</td>
<td>Enables AAA authentication for logins.</td>
</tr>
<tr>
<td><code>tacacs-server host</code>, on page 82</td>
<td>Specifies a TACACS+ host.</td>
</tr>
</tbody>
</table>
aaa authorization

To create a method list for authorization, use the `aaa authorization` command. To disable authorization for a function, use the `no` form of this command.

```plaintext
aaa authorization {commands | eventmanager | exec | network | subscriber | nacm} {default | list-name} 
{none | local | group {tacacs+ | radius | group-name}}
no aaa authorization {commands | eventmanager | exec | network | subscriber | nacm} {default | list-name}
```

### Syntax Description

- **commands**: Configures authorization for all EXEC shell commands.
- **eventmanager**: Applies an authorization method for authorizing an event manager (fault manager).
- **exec**: Configures authorization for an interactive (EXEC) session.
- **network**: Configures authorization for network services, such as PPP or Internet Key Exchange (IKE).
- **subscriber**: Sets the authorization lists for the subscriber.
- **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. This method of authorization is not available for command authorization.
- **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

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
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 EXEC mode commands a user issues. Command authorization attempts authorization for all EXEC mode commands.

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

- **EXEC authorization**—Applies authorization for starting an EXEC 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.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

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

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa authorization commands listname1 group tacacs+
```

<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>aaa accounting, on page 3</td>
<td>Creates a method list for accounting.</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

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

**Task ID**
- `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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.

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

**Syntax Description**

- `policy-name` Specifies the name of the password, in characters.
- `min-length` Specifies the minimum length of the password, in integer.
- `max-length` Specifies the maximum length of the password, 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.
- `lower-case` Specifies the number of lower case alphabets allowed in the password policy, in integer.
- `numeric` Specifies the number of numerals allowed in the password policy, in integer.
- `lifetime` Specifies the maximum lifetime for the password, the value of which is specified in integer, as years, months, days, hours, minutes or seconds.
- `min-char-change` Specifies the number of character change required between subsequent passwords, in integer.
- `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.
- `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.

**Command Default**

None

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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 8000 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 the *Configuring FIPS Mode* chapter in the *System Security Configuration Guide for Cisco 8000 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.

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

<table>
<thead>
<tr>
<th>Command Variables</th>
<th>Default Value</th>
<th>Maximum Value</th>
<th>Minimum Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>policy-name</code></td>
<td>None</td>
<td>253</td>
<td>1</td>
</tr>
<tr>
<td><code>max-length</code></td>
<td>253</td>
<td>253</td>
<td>2</td>
</tr>
<tr>
<td><code>min-length</code></td>
<td>2</td>
<td>253</td>
<td>2</td>
</tr>
<tr>
<td><code>special-char</code></td>
<td>0</td>
<td>253</td>
<td>0</td>
</tr>
<tr>
<td><code>upper-case</code></td>
<td>0</td>
<td>253</td>
<td>0</td>
</tr>
<tr>
<td><code>lower-case</code></td>
<td>0</td>
<td>253</td>
<td>0</td>
</tr>
<tr>
<td><code>numeric</code></td>
<td>0</td>
<td>253</td>
<td>0</td>
</tr>
</tbody>
</table>

For `lifetime`:
- `years`: 0 99 1
- `months`: 0 11 1
- `days`: 0 30 1
- `hours`: 0 23 1
- `minutes`: 0 59 1
- `seconds`: 0 59 1

<table>
<thead>
<tr>
<th>Command Variables</th>
<th>Default Value</th>
<th>Maximum Value</th>
<th>Minimum Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>min-char-change</code></td>
<td>4</td>
<td>253</td>
<td>0</td>
</tr>
<tr>
<td><code>authen-max-attempts</code></td>
<td>0</td>
<td>24</td>
<td>1</td>
</tr>
</tbody>
</table>

For `lockout-time`:
- `days`: 0 255 1
- `hours`: 0 23 1
- `minutes`: 0 59 1
- `seconds`: 0 59 1
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

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>username, on page 97</td>
<td></td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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(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**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>commands</td>
<td>Enables authorization on the selected lines for all commands.</td>
</tr>
<tr>
<td>exec</td>
<td>Enables authorization for an interactive XR EXEC mode session.</td>
</tr>
<tr>
<td>default</td>
<td>Applies the default method list, created with the aaa authorization command.</td>
</tr>
<tr>
<td>eventmanager</td>
<td>Sets eventmanager authorization method. This method is used for the embedded event manager.</td>
</tr>
<tr>
<td>list-name</td>
<td>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.</td>
</tr>
</tbody>
</table>

**Command Default**

Authorization is not enabled.

**Command Modes**

Line template configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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
```
deadtime (server-group configuration)

To configure the deadtime value at the RADIUS server group level, use the `deadtime` command in server-group configuration mode. To set deadtime to 0, use the `no` form of this command.

```
deadtime minutes
no deadtime
```

**Syntax Description**
- `minutes` 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.

**Command Default**
- Deadtime is set to 0.

**Command Modes**
- Server-group configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
The value of the deadtime set in the server groups overrides the deadtime that is configured globally. If the deadtime is omitted from the server group configuration, the value is inherited from the primary list. If the server group is not configured, the default value of 0 applies to all servers in the group. If the deadtime is set to 0, no servers are marked dead.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example specifies a one-minute deadtime for RADIUS server group `group1` when it has failed to respond to authentication requests for the `deadtime` command:

```
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 1.1.1.1 auth-port 1645 acct-port 1646
RP/0/RP0/CPU0:router(config-sg-radius)# server 2.2.2.2 auth-port 2000 acct-port 2001
RP/0/RP0/CPU0:router(config-sg-radius)# deadtime 1
```
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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.

```
{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.
- **maintenance**: Adds the user to the predefined 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 provisioning group.
- **retrieve**: Adds the user to the predefined 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.

**Note**

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 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 provisioning group.
- **retrieve**: Adds the user to the predefined 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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `group` command in username configuration mode. To access username configuration mode, use the `username`, on page 97 command in XR Config mode.

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.
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
```
**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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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.

**Syntax Description**

| **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**

<table>
<thead>
<tr>
<th><strong>Release</strong></th>
<th><strong>Modification</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### 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 (RADIUS)

To specify the authentication and encryption key that is used between the router and the RADIUS daemon running on the RADIUS server, use the `key (RADIUS)` command in RADIUS server-group private configuration mode.

```
key {0 clear-text-key | 7 encrypted-key}clear-text-key
no key {0 clear-text-key | 7 encrypted-key}clear-text-key
```

**Syntax Description**

- `0 clear-text-key` Specifies an unencrypted (cleartext) shared key.
- `7 encrypted-key` Specifies an encrypted shared key.
- `clear-text-key` Specifies an unencrypted (cleartext) user password.

**Command Default**

For submode `key` commands, the default is to use the `radius-server key` command in global configuration mode, if defined. If the global key is also not defined, the configuration is not complete.

**Command Modes**

RADIUS server-group private configuration

**Command History**

<table>
<thead>
<tr>
<th>Modification</th>
<th>Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>This command was introduced.</td>
<td>7.0.12</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

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

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

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa group server tacacs+, on page 17</td>
<td>Groups different RADIUS server hosts into distinct lists.</td>
</tr>
<tr>
<td>radius-server key, on page 43</td>
<td>Sets the authentication and encryption key for all RADIUS communications between the router and the RADIUS daemon.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>radius-server retransmit, on page 44</td>
<td>Specifies the number of times a RADIUS request is resent to a server if the server is not responding or is responding slowly.</td>
</tr>
<tr>
<td>server-private (RADIUS), on page 52</td>
<td>Configures the IP address of the private RADIUS server for the group server.</td>
</tr>
<tr>
<td>timeout (RADIUS), on page 94</td>
<td>Specifies the number of seconds the router waits for the RADIUS server to reply before retransmitting.</td>
</tr>
</tbody>
</table>
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-key auth-key}
no key {0 clear-text-key | 7 encrypted-key auth-key}
```

**Syntax Description**

- `0 clear-text-key` Specifies an unencrypted (cleartext) shared key.
- `7 encrypted-key auth-key` Specifies an encrypted shared key.

**Command Default**

None

**Command Modes**

TACACS host configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.

**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
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tacacs-server host</td>
<td>Specifies a TACACS+ host.</td>
</tr>
<tr>
<td>tacacs-server key</td>
<td>Globally sets the authentication encryption key used for all TACACS+ communications between the router and the TACACS+ daemon.</td>
</tr>
</tbody>
</table>
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**
--- | ---
7.0.12 | 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

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
<tr>
<td>tty-access</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

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

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

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="#">aaa authentication</a> on page 9</td>
<td>Creates a method list for authentication.</td>
</tr>
</tbody>
</table>
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.

```plaintext
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 7.0.12** This command was introduced.

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

When an EXEC process 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**
- `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
Building configuration...
username user1
password 7 141B1309

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>group (AAA), on page 25</td>
<td>Adds a user to a group.</td>
</tr>
<tr>
<td>usergroup, on page 96</td>
<td>Accesses user group configuration mode and configures a user group, associating it with a set of task groups.</td>
</tr>
<tr>
<td>username, on page 97</td>
<td>Accesses username configuration mode and configures a new user with a username, establishing a password and granting permissions for that user.</td>
</tr>
<tr>
<td>line</td>
<td>Enters line template configuration mode for the specified line template. For more information, see the Cisco IOS XR System Management Command Reference.</td>
</tr>
</tbody>
</table>
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 global configuration 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 to 120 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.

### 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

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### 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

- **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
```
<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>radius-server dead-criteria tries, on page 38</td>
<td>Specifies the number of consecutive timeouts that must occur on the router before the RADIUS server is marked as dead.</td>
</tr>
<tr>
<td></td>
<td>radius-server deadtime, on page 40</td>
<td>Defines the length of time, in minutes, for a RADIUS server to remain marked dead.</td>
</tr>
<tr>
<td></td>
<td>show radius dead-criteria, on page 70</td>
<td>Displays information for the dead-server detection criteria.</td>
</tr>
</tbody>
</table>
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. 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**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

- `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
```
<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>radius-server dead-criteria time, on page 36</td>
<td>Defines the length 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.</td>
</tr>
<tr>
<td></td>
<td>radius-server deadtime, on page 40</td>
<td>Defines the length of time, in minutes, for a RADIUS server to remain marked dead.</td>
</tr>
<tr>
<td></td>
<td>show radius dead-criteria, on page 70</td>
<td>Displays information for the dead-server detection criteria.</td>
</tr>
</tbody>
</table>
radius-server deadtime

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 XR Config mode. To set deadtime to 0, use the `no` form of this command.

```
radius-server deadtime minutes
```

**Syntax Description**

- `minutes` 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 7.0.12 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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

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

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

To specify a RADIUS server host, use the `radius-server host` command in XR Config mode. To delete the specified RADIUS host, use the `no` form of this command.

```
radius-server host 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. IPv6 address is not supported.
- `auth-port port-number`: (Optional) Specifies the User Datagram Protocol (UDP) destination port for authentication requests; the host is not used for authentication if set to 0. If unspecified, the port number defaults to 1645.
- `acct-port port-number`: (Optional) Specifies the UDP destination port for accounting requests; the host is not used for accounting if set to 0. If unspecified, the port number defaults to 1646.
- `timeout seconds`: (Optional) The time interval (in seconds) that the router waits for the RADIUS server to reply before retransmitting. This setting overrides the global value of the `radius-server timeout` command. If no timeout value is specified, the global value is used. Enter a value in the range from 1 to 1000. Default is 5.
- `retransmit retries`: (Optional) The number of times a RADIUS request is re-sent to a server, if that server is not responding or is responding slowly. This setting overrides the global setting of the `radius-server retransmit` command. If no retransmit value is specified, the global value is used. Enter a value in the range from 1 to 100. Default is 3.
- `key string`: (Optional) Specifies the authentication and encryption key used between the router and 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.

The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the `radius-server host` command syntax. This is because the leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key.

**Command Default**

No RADIUS host is specified; use global `radius-server` command values.

**Command Modes**

XR Config mode

**Command History**

- **Release 7.0.12** This command was introduced.

**Usage Guidelines**

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

If no host-specific timeout, retransmit, or key values are specified, the global values apply to each host.
This example shows how to establish the host with IP address 172.29.39.46 as the RADIUS server, use ports 1612 and 1616 as the authorization and accounting ports, set the timeout value to 6, set the retransmit value to 5, and set “rad123” as the encryption key, matching the key on the RADIUS server:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# radius-server host 172.29.39.46 auth-port 1612 acct-port 1616
        timeout 6 retransmit 5 key rad123
```

To use separate servers for accounting and authentication, use the zero port value as appropriate.
radius-server key

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-key clear-text-key}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 clear-text-key</td>
<td>Specifies an unencrypted (cleartext) shared key.</td>
</tr>
<tr>
<td>7 encrypted-key</td>
<td>Specifies a encrypted shared key.</td>
</tr>
<tr>
<td>clear-text-key</td>
<td>Specifies an unencrypted (cleartext) shared key.</td>
</tr>
</tbody>
</table>

### Command Default

The authentication and encryption key is disabled.

### Command Modes

XR Config mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### 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.

### 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

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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

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

**Task ID**
- `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

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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

- `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
```
To specify the number of times a RADIUS request is resent to a server if the server is not responding or is responding slowly, use the `retransmit` command in RADIUS server-group private configuration mode.

```
retransmit retries
no retransmit retries
```

**Syntax Description**

- `retries`: 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.

**Command Default**

The default value is 3.

**Command Modes**

RADIUS server-group private configuration

**Command History**

- **Release 7.0.12** This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Examples**

The following example shows how to set the retransmit value:

```
RP/0/CPU0:router# configure
RP/0/CPU0:router(config)# aaa group server radius group1
RP/0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 auth-port 300
RP/0/CPU0:router(config-sg-radius-private)# retransmit 100
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>aaa group server tacacs+</code></td>
<td>Groups different RADIUS server hosts into distinct lists.</td>
</tr>
<tr>
<td><code>server-private (RADIUS)</code></td>
<td>Configures the IP address of the private RADIUS server for the group server.</td>
</tr>
<tr>
<td><code>timeout (RADIUS), on page 94</code></td>
<td>Specifies the number of seconds the router waits for the RADIUS server to reply before retransmitting.</td>
</tr>
</tbody>
</table>
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.

### Command Default

No password is specified.

### Command Modes

- Username configuration
- Line template configuration

### Command History

**Release**

- Release 7.0.12

**Modification**

This command was introduced.

### Usage Guidelines

Type 10 encryption is applied as the default encryption type for the `secret` on Cisco IOS XR 64-bit operating systems.

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).
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.

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

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to establish the clear-text secret “lab” for the user `user2`:

```
Router# configure
Router(config)# username user2
Router(config-un)# secret 0 lab
Router(config-un)# commit
```

You can also see the examples and usage of the `username, on page 97` 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
$6$cqK7b1LdV8UEnqg6AFU3IL7KI/P_E4v/bhi_UaqLwY9R9seS9r98g6c59pbthr3m2t5g64jO83IQ7mea8wtyu5VM/aBFJgje/jxZG17h6xPcvJwF1
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**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>IP address of the RADIUS server host.</td>
</tr>
<tr>
<td>auth-port</td>
<td>(Optional) Specifies the User Datagram Protocol (UDP) destination port for authentication requests. The <code>port-number</code> 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.</td>
</tr>
<tr>
<td>acct-port</td>
<td>(Optional) Specifies the UDP destination port for accounting requests. The <code>port-number</code> 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.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.)
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 1.1.1.1 auth-port 1645 acct-port 1646
RP/0/RP0/CPU0:router(config-sg-radius)# server 2.2.2.2 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}
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hostname</td>
<td>Character string used to name the server host.</td>
</tr>
<tr>
<td>ip-address</td>
<td>IP address of the server host.</td>
</tr>
</tbody>
</table>

**Command Default**

None

**Command Modes**

TACACS+ server-group configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>IP address of the RADIUS server host.</td>
</tr>
<tr>
<td>auth-port</td>
<td>(Optional) Specifies the User Datagram Protocol (UDP) destination port for authentication requests. The <code>port-number</code> 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.</td>
</tr>
<tr>
<td>acct-port</td>
<td>(Optional) Specifies the UDP destination port for accounting requests. The <code>port-number</code> 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.</td>
</tr>
<tr>
<td>timeout</td>
<td>(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 <code>radius-server timeout</code> command. If no timeout is specified, the global value is used. The <code>seconds</code> argument specifies the timeout value in seconds. The range is from 1 to 1000. If no timeout is specified, the global value is used.</td>
</tr>
<tr>
<td>retransmit</td>
<td>(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 <code>radius-server transmit</code> command. The <code>retries</code> argument specifies the retransmit value. The range is from 1 to 100. If no retransmit value is specified, the global value is used.</td>
</tr>
<tr>
<td>key</td>
<td>(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 <code>radius-server key</code> command. If no key string is specified, the global value is used.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

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:

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

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# aaa group server radius group1
RP/0/RP0/CPU0:router(config)# server-private 10.1.1.1 auth-port 300
RP/0/RP0/CPU0:router(config)# exit
RP/0/RP0/CPU0:router(config)# server-private 10.2.2.2 auth-port 300
RP/0/RP0/CPU0:router(config)# exit
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 {hostname ip-address} [port port-number] [timeout seconds] [key string]
no server-private {hostname ip-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.

- `port`  
  (Optional) Specifies a server port number. This option overrides the default, which is port 49. Valid port numbers range from 1 to 65535.

- `timeout`  
  (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`  
  (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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.
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

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.

```plaintext
show aaa {ikegroup ikegroup-name | login trace | usergroup [usergroup-name] | trace | userdb [username] | task supported | taskgroup [{root-lr | netadmin | operator | sysadmin | root-system | service-admin | cisco-support | askgroup-name]}
```

**Syntax Description**

- `ikegroup` Displays details for all IKE groups.
  - `ikegroup-name` (Optional) IKE group whose details are to be displayed.

- `login trace` Displays trace data for login subsystem.

- `usergroup` Displays details for all user groups.
  - `root-lr` (Optional) Usergroup name.
  - `netadmin` (Optional) Usergroup name.
  - `operator` (Optional) Usergroup name.
  - `sysadmin` (Optional) Usergroup name.
  - `root-system` (Optional) Usergroup name.

- `cisco-support` (Optional) Usergroup name.
  - `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 supported` Displays all AAA task IDs available.

- `taskgroup` Displays details for all task groups.
  - `taskgroup-name` (Optional) Task group whose details are to be displayed.

**Command Default**

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

**Command Modes**

`EXEC`
Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

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

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read</td>
</tr>
</tbody>
</table>

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 WRITE EXECUTE DEBUG
  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
```
show aaa
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:         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-mgmt 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 hdle : READ WRITE EXECUTE DEBUG
Task:         hdle 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 pos-mgmt dpt : 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
```
Authentication, Authorization, and Accounting Commands

The following sample output is from `show aaa` command with the `userdb` keyword:

RP/0/RP0/CPU0:router# show aaa userdb

Username lab
User group root-lr
User group cisco-support

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
bdsl
bfd
bqp
boot
bundle
cdp
cfg
cisco-support
config-mgmt
config-services
crypto
diag
disallowed
drivers
eigrp
ext-access
fabric
fault-mgmt
filesystem
firewall
fr
hdlc
host-services
hsrp
interface
inventory
ip-services
ipv4
Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show user, on page 77</td>
<td>Displays task IDs enabled for the currently logged-in user.</td>
</tr>
</tbody>
</table>
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**: 7.0.12
  - **Modification**: 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**: aaa
- **Operation**: 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
```
Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa password-policy, on page 18</td>
<td>Defines the FIPS-compliant AAA password security policy.</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read</td>
</tr>
</tbody>
</table>

**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 1: show radius accounting Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Server IP address/UDP destination port for authentication requests; UDP destination port for accounting requests.</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read</td>
</tr>
</tbody>
</table>

**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 2: show radius authentication Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Server IP address/UDP destination port for authentication requests; UDP destination port for accounting requests.</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

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

**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: 1.1.1.1/1645/1646  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: 2.2.2.2/1645/1646  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
```
This table describes the significant fields shown in the display.

**Table 3: show radius Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Server IP address/UDP destination port for authentication requests/UDP destination port for accounting requests.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Number of seconds the router waits for a server host to reply before timing out.</td>
</tr>
<tr>
<td>Retransmit limit</td>
<td>Number of times the Cisco IOS XR software searches the list of RADIUS server hosts before giving up.</td>
</tr>
</tbody>
</table>
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 7.0.12** This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **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>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Server IP address/UDP destination port for authentication requests/UDP destination port for accounting requests.</td>
</tr>
<tr>
<td>Timeout</td>
<td>Number of seconds the router waits for a server host to reply before timing out.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Retransmits</td>
<td>Number of times Cisco IOS XR software searches the list of RADIUS server hosts before giving up.</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.

**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 1.1.1.1/1645/1646
    Server 2.2.2.2/1645/1646

Server group 'radgrp1' has 2 server(s)
    Dead time: 0 minute(s) (inherited from global)
Contains 2 server(s)
    Server 1.1.1.1/1645/1646
    Server 2.2.2.2/1645/1646

Server group 'radgrp-priv' has 1 server(s)
    Dead time: 0 minute(s) (inherited from global)
Contains 1 server(s)
    Server 3.3.3.3/1645/1646 [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 1.1.1.1/1645/1646
      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 2.2.2.2/1645/1646
      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 3.3.3.3/1645/1646 [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 5: show radius server-groups Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Server IP address/UDP destination port for authentication requests/UDP destination port for accounting requests.</td>
</tr>
</tbody>
</table>
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.

**Syntax Description**

This command has no keywords or arguments.

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

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

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

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

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

For IPv4 IP addresses:
Server:1.1.1.1/21212 opens=0 closes=0 aborts=0 errors=0
   packets in=0 packets out=0
   status=up single-connect=false
Server:2.2.2.2/21232 opens=0 closes=0 aborts=0 errors=0
   packets in=0 packets out=0
   status=up single-connect=false

For IPv6 IP addresses:
Server: 1.2.3.5/49 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 6: show tacacs Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Server IP address.</td>
</tr>
<tr>
<td>opens</td>
<td>Number of socket opens to the external server.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>------------------------------------------------------------</td>
</tr>
<tr>
<td>closes</td>
<td>Number of socket closes to the external server.</td>
</tr>
<tr>
<td>aborts</td>
<td>Number of tacacs requests that have been terminated midway.</td>
</tr>
<tr>
<td>errors</td>
<td>Number of error replies from the external server.</td>
</tr>
<tr>
<td>packets in</td>
<td>Number of TCP packets that have been received from the external server.</td>
</tr>
<tr>
<td>packets out</td>
<td>Number of TCP packets that have been sent to the external server.</td>
</tr>
</tbody>
</table>
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.

**Syntax Description**

This command has no keywords or arguments.

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.

**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 12.26.25.61/23456
   Server 12.26.49.12/12345
   Server 12.26.49.12/9000
   Server 12.26.25.61/23432
   Server 5.5.5.5/23456
   Server 1.1.1.1/49
Server group 'tac100' has 1 servers
   Server 12.26.49.12
```

This table describes the significant fields shown in the display.

**Table 7: show tacacs server-groups Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Server IP address.</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

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

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
<td>---</td>
</tr>
</tbody>
</table>

**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:

```
```
Username: lab
Groups: root-system
Authenticated using method local
User lab has the following Task ID(s):

Task: aaa : READ WRITE EXECUTE DEBUG
Task: atm : READ WRITE EXECUTE DEBUG
Task: basic-services : 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: hdic : 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

Authentication, Authorization, and Accounting Commands
The following sample output displays the tasks and indicates which tasks are reserved from the `show user` command:

```text
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:           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:           hrsp : 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
```
show user

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: sbr : 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
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.

This command has no keywords or arguments.

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

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.

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
```
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 [port port-number] [timeout seconds] [key [0 | 7] auth-key] [single-connection]
no tacacs-server host host-name [port port-number]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host-name</td>
<td>Host or domain name or IP address of the TACACS+ server.</td>
</tr>
<tr>
<td>port port-number</td>
<td>(Optional) Specifies a server port number. This option overrides the default, which is port 49. Valid port numbers range from 1 to 65535.</td>
</tr>
<tr>
<td>timeout seconds</td>
<td>(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 <code>tacacs-server timeout</code> command for this server only. The valid timeout range is from 1 to 1000 seconds. Default is 5.</td>
</tr>
<tr>
<td>key [0</td>
<td>7] auth-key</td>
</tr>
<tr>
<td></td>
<td>(Optional) Entering 0 specifies that an unencrypted (clear-text) key follows.</td>
</tr>
<tr>
<td></td>
<td>(Optional) Entering 7 specifies that an encrypted key follows.</td>
</tr>
<tr>
<td></td>
<td>The <code>auth-key</code> argument specifies the unencrypted key between the AAA server and the TACACS+ server.</td>
</tr>
<tr>
<td></td>
<td>Note: You can use this parameter only in the config-tacacs-host sub-mode.</td>
</tr>
<tr>
<td>single-connection</td>
<td>(Optional) Multiplexes all TACACS+ requests to this server over a single TCP connection. By default, a separate connection is used for each session.</td>
</tr>
<tr>
<td></td>
<td>Note: You can use this parameter only in the config-tacacs-host sub-mode.</td>
</tr>
</tbody>
</table>

### 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.

### Command Modes

XR Config mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
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.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

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
```

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
```
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**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ipv4</code></td>
<td>Specifies the dscp bit for the IPv4 packets.</td>
</tr>
<tr>
<td><code>dscp</code></td>
<td>Sets the DSCP in the IP header.</td>
</tr>
</tbody>
</table>

`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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

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

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

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

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

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.

**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 HundredGigabitEthernet 0/0/0/29 vrf abc
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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 ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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

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

#### Command Default

Five predefined user groups are available by default.

#### Command Modes

XR Config mode

#### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

#### 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.

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.
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td></td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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 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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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.

**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
```
timeout (RADIUS)

To specify the number of seconds the router waits for the RADIUS server to reply before retransmitting, use the `timeout` command in RADIUS server-group private 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**

RADIUS server-group private configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

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

```
RP/0/CPU0:router# configure
RP/0/CPU0:router(config)# aaa group server radius group1
RP/0/CPU0:router(config-sg-radius)# server-private 10.1.1.1 auth-port 300
RP/0/CPU0:router(config-sg-radius-private)# timeout 500
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>radius-server timeout, on page 45</td>
<td>Sets the interval for which a router waits for a server host to reply before timing out.</td>
</tr>
<tr>
<td>radius-server retransmit, on page 44</td>
<td>Specifies the number of times a RADIUS request is resent to a server if the server is not responding or is responding slowly.</td>
</tr>
<tr>
<td>server-private (RADIUS), on page 52</td>
<td>Configures the IP address of the private RADIUS server for the group server.</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**
  - 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
```
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

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### 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` 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

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### 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, associate a password policy with the user, grant permissions for the user, and to enter username configuration mode, use the `username` command in XR Config mode. To delete a user from the database, use the `no` form of this command.

```
username user-name [{password [0 | 7] password | password-policy policy-name password [ { 0 | 7 } ] password | secret [ { 0 | 5 | 8 | 9 | 10 } password | group usergroup-name }]}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>user-name</td>
<td>Name of the user. The <code>user-name</code> 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.</td>
</tr>
<tr>
<td>password</td>
<td>(Optional) Enables a password to be created for the named user.</td>
</tr>
<tr>
<td>0</td>
<td>(Optional) Specifies that an unencrypted (clear-text) password follows. The password will be encrypted for storage in the configuration using a Cisco proprietary encryption algorithm.</td>
</tr>
<tr>
<td>7</td>
<td>(Optional) Specifies that an encrypted password follows.</td>
</tr>
<tr>
<td>password</td>
<td>(Optional) 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.</td>
</tr>
<tr>
<td>password-policy</td>
<td>Applies a particular password policy to the user.</td>
</tr>
<tr>
<td>policy-name</td>
<td>Specifies the name of the password policy.</td>
</tr>
<tr>
<td></td>
<td>This policy name has to be configured prior to applying this policy to the user.</td>
</tr>
</tbody>
</table>
**secret**

(Optional) Enables an MD5-secured password to be created for the named user.

**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.

**5**

(Optional) Specifies that an encrypted password follows.

**8**

(Optional) Specifies that Type 8 password that uses SHA256 hashing algorithm follows.

**9**

(Optional) Specifies that Type 9 password that uses scrypt hashing algorithm follows.

**10**

(Optional) Specifies that Type 10 password that uses SHA512 hashing algorithm follows.

**group**

(Optional) Enables a named user to be associated with a user group.

**usergroup-name**

(Optional) Name of a user group as defined with the `usergroup` command.

---

**Command Default**

No usernames are defined in the system.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

- A user is never allowed to have cisco-support privileges as the only group.
- Type 10 (SHA512) is the default password type for the `secret` configuration.

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. You can display configured usernames in configuration mode by router(config): **do show run username**.

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 **aaa authentication** 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.

---

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

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

```
Router# config
Router(config)# username user1
Router(config-un)# ?
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>clear</td>
<td>Clear the uncommitted configuration</td>
</tr>
<tr>
<td>commit</td>
<td>Commit the configuration changes to running</td>
</tr>
<tr>
<td>describe</td>
<td>Describe a command without taking real actions</td>
</tr>
<tr>
<td>do</td>
<td>Run an exec command</td>
</tr>
<tr>
<td>exit</td>
<td>Exit from this submode</td>
</tr>
<tr>
<td>group</td>
<td>User group in which this user will be a member of</td>
</tr>
<tr>
<td>no</td>
<td>Negate a command or set its defaults</td>
</tr>
<tr>
<td>password</td>
<td>Specify the password for the user</td>
</tr>
<tr>
<td>pwd</td>
<td>Commands used to reach current submode</td>
</tr>
<tr>
<td>root</td>
<td>Exit to the XR Config mode</td>
</tr>
<tr>
<td>secret</td>
<td>Specify the secure password for the user</td>
</tr>
</tbody>
</table>
The following example shows how to establish the clear-text password `password1` for the user name `user1`:

```bash
Router# configure
Router(config)# username user1
Router(config-un)# password 0 password1
```

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 47 command.

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

```bash
Router# configure
Router(config)# username user8 secret 8
$8$ZYKGl1dZIw73D1$1UWJOqTLoMyExhsNKO5vMtVCOYguM5ajXf4uGeQj6I
Router(config-un)# commit
```

This example shows how to configure Type 9 password:

```bash
Router# configure
Router(config)# username user9 secret 9
$9$/rIQLb3p1rBL$9s2fLWKFY6B/kApXkkXnIqbFAHpRZkPEoh3WqGbvvQ
Router(config-un)# commit
```

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

```bash
Router# configure
Router(config)# username user10 secret 10
$6$9UvJidvzTEqgkAFO$3C1ELi/F.E4v/Hl.UsqGwX80s9Ez9pG6c5pzhMjM2tgM4jOhaQ7ma1wyhu5VM/aRFJo/jeZGl7hoXPrvJMF1
Router(config-un)# 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 | maintenance | netadmin | operator | provisioning | retrieve | root-lr | serviceadmin | sysadmin}
```

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

**Syntax Description**

<table>
<thead>
<tr>
<th><code>usergroup-name</code></th>
<th>Name of the user group. The <code>usergroup-name</code> argument can be only one word. Spaces and quotation marks are not allowed.</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cisco-support</code></td>
<td>Specifies that users logging in through the line are given Cisco support personnel privileges.</td>
</tr>
<tr>
<td><code>maintenance</code></td>
<td>Specifies that users logging in through the line are given SCAPA maintenance privileges.</td>
</tr>
<tr>
<td><code>netadmin</code></td>
<td>Specifies that users logging in through the line are given network administrator privileges.</td>
</tr>
<tr>
<td><code>operator</code></td>
<td>Specifies that users logging in through the line are given operator privileges.</td>
</tr>
<tr>
<td><code>provisioning</code></td>
<td>Specifies that users logging in through the line are given SCAPA provisioning privileges.</td>
</tr>
<tr>
<td><code>retrieve</code></td>
<td>Specifies that users logging in through the line are given SCAPA retrieve privileges.</td>
</tr>
<tr>
<td><code>root-lr</code></td>
<td>Specifies that users logging in through the line are given root logical router (LR) privileges.</td>
</tr>
<tr>
<td><code>serviceadmin</code></td>
<td>Specifies that users logging in through the line are given service administrator group privileges.</td>
</tr>
<tr>
<td><code>sysadmin</code></td>
<td>Specifies that users logging in through the line are given system administrator privileges.</td>
</tr>
</tbody>
</table>
**users group**

**Command Default**

None

**Command Modes**

Line template configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**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
```
The vrf (RADIUS) command is used to configure the Virtual Private Network (VPN) routing and forwarding (VRF) reference of an AAA RADIUS server group. To enable server groups to use the global (default) routing table, use the `no` form of this command.

**vrf**

```plaintext
vrf vrf-name
no vrf vrf-name
```

**Syntax Description**
- `vrf-name`: Name assigned to a VRF.

**Command Default**
- The default VRF is used.

**Command Modes**
- RADIUS server-group configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
- Use the `vrf` command to specify a VRF for an AAA RADIUS server group and enable dial-up users to use AAA servers in different routing domains.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to use the `vrf` command:

```
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)# vrf vrf1
```
vrf (TACACS+)

To configure the Virtual Private Network (VPN) routing and forwarding (VRF) reference of an AAA TACACS+ server group, use the `vrf` command in TACACS+ server-group configuration mode. To enable server groups to use the global (default) routing table, use the `no` form of this command.

```
vrf vrf-name
no vrf vrf-name
```

**Syntax Description**

`vrf-name` Name assigned to a VRF.

**Command Default**
The default VRF is used.

**Command Modes**
TACACS+ server-group configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 7.0.12</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `vrf` command to specify a VRF for an AAA TACACS+ server group and enable dial-up users to use AAA servers in different routing domains.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>aaa</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
This example shows how to use the `vrf` command:

```
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 9.27.10.6
RP/0/RP0/CPU0:router(config-sg-tacacs+)# vrf abc
```