



# Authorization Commands

---

This chapter describes the commands used to configure authentication, authorization, and accounting (AAA) authorization. AAA authorization enables you to limit the services available to a user. When AAA authorization is enabled, the network access server uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. Once this is done, the user will be granted access to a requested service only if the information in the user profile allows it.

For information on how to configure authorization using AAA, refer to the “Configuring Authorization” chapter in the *Security Configuration Guide*. For configuration examples using the commands in this chapter, refer to the “Authorization Configuration Examples” section located at the end of the “Configuring Authorization” chapter in the *Security Configuration Guide*.

# aaa authorization

Use the **aaa authorization** global configuration command to set parameters that restrict a user’s network access. Use the **no** form of this command to disable authorization for a function.

```
aaa authorization {network | exec | command level} method
no aaa authorization {network | exec | command level}
```

## Syntax Description

<b>network</b>	Runs authorization for all network-related service requests, including SLIP, PPP, PPP NCPs, and ARA protocol.
<b>exec</b>	Runs authorization to determine if the user is allowed to run an EXEC shell. This facility might return user profile information such as <b>autocommand</b> information.
<b>command</b>	Runs authorization for all commands at the specified privilege level.
<i>level</i>	Specific command level that should be authorized. Valid entries are 0 through 15.
<i>method</i>	One of the keywords in Table 6.

## Default

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

## Command Mode

Global configuration

## Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

---

**Note** There are five commands associated with privilege level 0: **disable**, **enable**, **exit**, **help**, and **logout**. If you configure AAA authorization for a privilege level greater than 0, these five commands will not be included in the privilege level command set.

---

Use the **aaa authorization** command to create at least one, and up to four, authorization methods that can be used when a user accesses the specified function. Method keywords are described in Table 6.

---

**Note** This command, along with **aaa accounting**, replaces the **tacacs-server** suite of commands in previous versions of TACACS.

---

The additional methods of authorization are used only if the previous method returns an error, not if it fails. Specify **none** as the final method in the command line to have authorization succeed even if all methods return an error.

If authorization is not specifically set for a function, the default is **none** and no authorization is performed.

**Table 6 AAA Authorization Methods**

Keyword	Description
<b>tacacs+</b>	Requests authorization information from the TACACS+ server.
<b>if-authenticated</b>	Allows the user to access the requested function if the user is authenticated.
<b>none</b>	No authorization is performed.
<b>local</b>	Uses the local database for authorization.
<b>radius</b>	Uses RADIUS to get authorization information.
<b>krb5-instance</b>	Uses the instance defined by the <b>Kerberos instance map</b> command.

The authorization command causes a request packet containing a series of attribute value pairs to be sent to the RADIUS or TACACS daemon as part of the authorization process. The daemon can do one of the following:

- Accept the request as is
- Make changes to the request
- Refuse the request and refuse authorization

For a list of supported RADIUS attributes, refer to the “RADIUS Attributes” appendix in the *Security Configuration Guide*. For a list of supported TACACS+ AV pairs, refer to the “TACACS+ AV Pairs” appendix in the *Security Configuration Guide*.

## Examples

The following example specifies that TACACS+ authorization is used for all network-related requests. If this authorization method returns an error (if the TACACS+ server cannot be contacted), no authorization is performed and the request succeeds.

```
aaa authorization network tacacs+ none
```

The following example specifies that TACACS+ authorization is run for level 15 commands. If this authorization method returns an error (if the TACACS+ server cannot be contacted), no authorization is performed and the request succeeds.

```
aaa authorization command 15 tacacs+ none
```

## Related Commands

You can use the master indexes or search online to find documentation of related commands.

**aaa accounting**  
**aaa authorization reverse-access**  
**aaa new-model**

## aaa authorization config-commands

To disable AAA configuration command authorization in the EXEC mode, use the **no** form of the **aaa authorization config-commands** global configuration command. Use the standard form of this command to reestablish the default created when the **aaa authorization command level method** command was issued.

**aaa authorization config-commands**  
**no aaa authorization config-commands**

### Syntax Description

This command has no arguments or keywords.

### Default

After the **aaa authorization command level method** has been issued, this command is enabled by default—meaning that all configuration commands in the EXEC mode will be authorized.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

If **aaa authorization command level method** is enabled, all commands, including configuration commands, are authorized by AAA using the method specified. Because there are configuration commands that are identical to some EXEC-level commands, there can be some confusion in the authorization process. Using **no aaa authorization config-commands** stops the network access server not from attempting configuration command authorization.

After the **no** form of this command has been entered, AAA authorization of configuration commands is completely disabled. Care should be taken before entering the **no** form of this command because it potentially reduces the amount of administrative control on configuration commands.

Use the **aaa authorization config-commands command** if, after using the **no** form of this command, you need to reestablish the default set by the **aaa authorization command level method** command.

### Example

The following example specifies that TACACS+ authorization is run for level 15 commands and that AAA authorization of configuration commands is disabled:

```
aaa new-model
aaa authorization command 15 tacacs+ none
no aaa authorization config-commands
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**aaa authorization**

## aaa authorization reverse-access

To configure a NAS to request authorization information from a security server before allowing a user to establish a reverse Telnet session, use the **aaa authorization reverse-access** global configuration command. Use the **no** form of this command to restore the default value for this command.

```
aaa authorization reverse-access {radius | tacacs+}  
no aaa authorization reverse-access {radius | tacacs+}
```

### Syntax Description

<b>radius</b>	Specifies that the NAS will request authorization from a RADIUS security server before allowing a user to establish a reverse Telnet session.
<b>tacacs+</b>	Specifies that the NAS will request authorization from a TACACS+ security server before allowing a user to establish a reverse Telnet session.

### Default

The default for this command is disabled, meaning that authorization for reverse Telnet is not requested.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.3.

Telnet is a standard terminal emulation protocol used for remote terminal connection. Normally, you log in to a network access server (NAS) (typically through a dialup connection) and then use Telnet to access other network devices from that NAS. There are times, however, when it is necessary to establish a reverse Telnet session. In reverse Telnet sessions, the Telnet connection is established in the opposite direction—from inside a network to a NAS on the network periphery to gain access to modems or other devices connected to that NAS. Reverse Telnet is used to provide users with dialout capability by allowing them to Telnet to modem ports attached to a NAS.

It is important to control access to ports accessible through reverse Telnet. Failure to do so could, for example, allow unauthorized users free access to modems where they can trap and divert incoming calls or make outgoing calls to unauthorized destinations.

Authentication during reverse Telnet is performed through the standard AAA login procedure for Telnet. Typically the user has to provide a username and password to establish either a Telnet or reverse Telnet session. This command provides an additional (optional) level of security by requiring authorization in addition to authentication. When this command is enabled, reverse Telnet authorization can use RADIUS or TACACS+ to authorize whether or not this user is allowed reverse Telnet access to specific asynchronous ports, after the user successfully authenticates through the standard Telnet login procedure.

## Examples

The following example causes the NAS to request authorization information from a TACACS+ security server before allowing a user to establish a reverse Telnet session:

```
aaa new-model
aaa authentication login default tacacs+
aaa authorization reverse-access tacacs+
!
tacacs-server host 172.31.255.0
tacacs-server timeout 90
tacacs-server key goaway
```

The lines in this sample TACACS+ reverse Telnet authorization configuration are defined as follows:

- The **aaa new-model** command enables AAA.
- The **aaa authentication login default tacacs+** command specifies TACACS+ as the default method for user authentication during login.
- The **aaa authorization reverse-access tacacs+** specifies TACACS+ as the method for user authorization when trying to establish a reverse Telnet session.
- The **tacacs-server host** command identifies the TACACS+ server.
- The **tacacs-server timeout** command sets the interval of time that the NAS waits for the TACACS+ server to reply.
- The **tacacs-server key** command defines the encryption key used for all TACACS+ communications between the NAS and the TACACS+ daemon.

The following example configures a generic TACACS+ server to grant a user, “jim,” reverse Telnet access to port tty2 on the NAS named godzilla and to port tty5 on the NAS named gamera:

```
user = jim
login = cleartext lab
service = raccess {
    port#1 = godzilla/tty2
    port#2 = gamera/tty5
}
```

---

**Note** In this example, “godzilla” and “gamera” are the configured hostnames of network access servers, not DNS names or alias.

---

The following example configures the TACACS+ server (CiscoSecure) to authorize a user named Jim for reverse Telnet:

```
user = jim
profile_id = 90
profile_cycle = 1
member = Tacacs_Users
service=shell {
    default cmd=permit
}
service=raccess {
    allow "c2511e0" "tty1" ".*"
    refuse ".*" ".*" ".*"
    password = clear "goaway"
```

---

**Note** CiscoSecure only supports reverse Telnet using the command line interface in versions 2.1(x) through version 2.2(1).

---

An empty “service=raccess { }” clause permits a user to have unconditional access to NAS ports for reverse Telnet. If no “service=raccess” clause exists, the user is denied access to any port for reverse Telnet.

For more information about configuring TACACS+, refer to the “Configuring TACACS+” chapter in the Cisco IOS Release 11.3 *Security Configuration Guide*. For more information about configuring CiscoSecure, refer to the *CiscoSecure Access Control Server User Guide*, version 2.1(2) or greater.

The following example causes the NAS to request authorization from a RADIUS security server before allowing a user to establish a reverse Telnet session:

```
aaa new-model
aaa authentication login default radius
aaa authorization reverse-access radius
!
radius-server host 172.31.255.0
radius-server key go away
```

The lines in this sample RADIUS reverse Telnet authorization configuration are defined as follows:

- The **aaa new-model** command enables AAA.
- The **aaa authentication login default radius** command specifies RADIUS as the default method for user authentication during login.
- The **aaa authorization reverse-access radius** specifies RADIUS as the method for user authorization when trying to establish a reverse Telnet session.
- The **radius-server host** command identifies the RADIUS server.
- The **radius-server key** command defines the encryption key used for all RADIUS communications between the NAS and the RADIUS daemon.

The following example configures the RADIUS server to grant a user named “jim” reverse Telnet access at port tty2 on NAS godzilla:

```
Password = "goaway"
User-Service-Type = Shell-User
cisco-avpair = "raccess:port#1=godzilla/tty2"
```

An empty “raccess:port#1=nasname1/tty2” clause permits a user to have unconditional access to NAS ports for reverse Telnet. If no “raccess:port#1=nasname1/tty2” clause exists, the user is denied access to any port for reverse Telnet.

For more information about configuring RADIUS, refer to the “Configuring RADIUS” chapter in the Cisco IOS Release 11.3 *Security Configuration Guide*.

## Related Commands

**aaa authorization**

## aaa new-model

To enable the AAA access control model, use the **aaa new-model** global configuration command. Use the **no** form of this command to disable this functionality.

**aaa new-model**  
**no aaa new-model**

### Syntax Description

This command has no arguments or keywords.

### Default

AAA is not enabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

This command enables the AAA access control system. After you have enabled AAA, TACACS and Extended TACACS commands are no longer available. If you initialize AAA functionality and later decide to use TACACS or extended TACACS, issue the **no** version of this command then enable the version of TACACS that you want to use.

### Example

The following example initializes AAA:

```
aaa new-model
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**aaa accounting**  
**aaa authentication arap**  
**aaa authentication enable default**  
**aaa authentication local-override**  
**aaa authentication login**  
**aaa authentication ppp**  
**aaa authorization**  
**tacacs-server key**