



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

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

```
aaa authorization { network | exec | commands level | reverse-access } { default | list-name }
    [method1 [method2...]]
```

```
no aaa authorization { network | exec | commands level | reverse-access }
```

Syntax Description

network	Runs authorization for all network-related service requests, including SLIP, PPP, PPP NCPs, and ARA.
exec	Runs authorization to determine if the user is allowed to run an EXEC shell. This facility might return user profile information such as autocommand information.
commands	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.
reverse-access	Runs authorization for reverse access connections, such as reverse Telnet.
default	Uses the listed authorization methods that follow this argument as the default list of methods for authorization.
<i>list-name</i>	Character string used to name the list of authorization methods.
<i>method1</i> [<i>method2...</i>]	One of the keywords listed in Table 399.

Defaults

Authorization is disabled for all actions (equivalent to the method keyword **none**). If the **aaa authorization** command for a particular authorization type is issued without a named method list specified, the default method list is automatically applied to all interfaces or lines (where this authorization type applies) except those that have a named method list explicitly defined. (A defined method list overrides the default method list.) If no default method list is defined, then no authorization takes place.

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.

Usage Guidelines



Note

This command cannot be used with TACACS or extended TACACS.

Use the **aaa authorization** command to enable authorization and to create named methods lists, defining authorization methods that can be used when a user accesses the specified function. 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 simply a named list describing the authorization methods to be queried (such as RADIUS or TACACS+), in sequence. Method lists enable you to designate one or more security protocols to be used for authorization, thus ensuring a backup system in case the initial method fails. Cisco IOS software uses the first method listed to authorize users for specific network services; if that method fails to respond, the Cisco IOS software selects the next method listed in the method list. This process continues until there is successful communication with a listed authorization method, or all methods defined are exhausted.

**Note**

The Cisco IOS software attempts authorization with the next listed method only when there is no response 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.

Use the **aaa authorization** command to create a list by entering the *list-name* and the *method*, where *list-name* is any character string used to name this list (excluding all method names) and *method* identifies the list of authorization method(s) tried in the given sequence.

Method keywords are described in Table 399.

Table 399 *aaa authorization Methods*

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

Cisco IOS software supports the following six methods for authorization:

- **TACACS+**—The network access server exchanges authorization information with the TACACS+ security daemon. TACACS+ authorization defines specific rights for users by associating attribute-value (AV) pairs, which are stored in a database on the TACACS+ security server, with the appropriate user.
- **If-Authenticated**—The user is allowed to access the requested function provided the user has been authenticated successfully.
- **None**—The network access server does not request authorization information; authorization is not performed over this line/interface.
- **Local**—The router or access server consults its local database, as defined by the **username** command, to authorize specific rights for users. Only a limited set of functions can be controlled via the local database.

- **RADIUS**—The network access server requests authorization information from the RADIUS security server. RADIUS authorization defines specific rights for users by associating attributes, which are stored in a database on the RADIUS server, with the appropriate user.
- **Kerberos Instance Map**—The network access server uses the instance defined by the **kerberos instance map** command for authorization.

Method lists are specific to the type of authorization being requested. AAA supports four different types of authorization:

- **Network**—Applies to network connections. This can include a PPP, SLIP, or ARA connection.
- **EXEC**—Applies to the attributes associated with a user EXEC terminal session.
- **Commands**—Applies to the EXEC mode commands a user issues. Command authorization attempts authorization for all EXEC mode commands, including global configuration commands, associated with a specific privilege level.
- **Reverse Access**—Applies to reverse Telnet sessions.

When you create a named method list, you are defining a particular list of authorization methods for the indicated authorization type.

Once defined, method lists must be applied to specific lines or interfaces before any of the defined methods will be performed.

The authorization command causes a request packet containing a series of AV 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+ Attribute-Value Pairs” appendix in the *Security Configuration Guide*.



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.

Examples

The following example defines the network authorization method list named scoobee, which specifies that RADIUS authorization will be used on serial lines using PPP. If the RADIUS server fails to respond, then local network authorization will be performed.

```
aaa authorization network scoobee radius local
```

Related Commands

Command	Description
aaa accounting	Enables AAA accounting of requested services for billing or security purposes.
aaa new-model	Enables the AAA access control 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 commands level method1** command was issued.

aaa authorization config-commands

no aaa authorization config-commands

Syntax Description

This command has no arguments or keywords.

Defaults

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

Command Modes

Global configuration

Command History

Release	Modification
11.2	This command was introduced.

Usage Guidelines

If **aaa authorization commands 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 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 commands level method** command.

Examples

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

Command	Description
aaa authorization	Sets parameters that restrict network access to a user.

aaa authorization reverse-access

To configure a network access server 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	radius	tacacs+
	Specifies that the network access server will request authorization from a RADIUS security server before allowing a user to establish a reverse Telnet session.	Specifies that the network access server will request authorization from a TACACS+ security server before allowing a user to establish a reverse Telnet session.

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

Command Modes Global configuration

Command History	Release	Modification
	11.3	This command was introduced.

Usage Guidelines Telnet is a standard terminal emulation protocol used for remote terminal connection. Normally, you log in to a network access server (typically through a dialup connection) and then use Telnet to access other network devices from that network access server. 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 network access server on the network periphery to gain access to modems or other devices connected to that network access server. Reverse Telnet is used to provide users with dialout capability by allowing them to Telnet to modem ports attached to a network access server.

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 network access server 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 network access server waits for the TACACS+ server to reply.
- The **tacacs-server key** command defines the encryption key used for all TACACS+ communications between the network access server 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 network access server named “site1” and to port tty5 on the network access server named camera:

```
user = jim
  login = cleartext lab
  service = raccess {
    port#1 = site1/tty2
    port#2 = site2/tty5
```



Note

In this example, “site1” and “site2” are the configured host names 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 network access server 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 *Security Configuration Guide*. For more information about configuring CiscoSecure, refer to the *CiscoSecure Access Control Server User Guide*, version 2.1(2) or later.

The following example causes the network access server 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 network access server and the RADIUS daemon.

The following example configures the RADIUS server to grant a user named "jim" reverse Telnet access at port tty2 on network access server site1:

```
Password = "goaway"
User-Service-Type = Shell-User
cisco-avpair = "raccess:port#1=site1/tty2"
```

The syntax "raccess:port=any/any" permits a user to have unconditional access to network access server ports for reverse Telnet. If no "raccess:port={nasname}/{tty number}" clause exists in the user profile, the user is denied access to reverse Telnet on all ports.

For more information about configuring RADIUS, refer to the "Configuring RADIUS" chapter in the *Security Configuration Guide*.

authorization

To enable AAA authorization for a specific line or group of lines, use the **authorization** line configuration command. Use the **no** form of this command to disable authorization.

authorization { **arap** | **commands** *level* | **exec** | **reverse-access** } [**default** | *list-name*]

no authorization { **arap** | **commands** *level* | **exec** | **reverse-access** } [**default** | *list-name*]

Syntax Description	
arap	Enables authorization for line(s) configured for AppleTalk Remote Access (ARA) protocol.
commands	Enables authorization on the selected line(s) for all commands at the specified privilege level.
<i>level</i>	Specific command level to be authorized. Valid entries are 0 through 15.
exec	Enables authorization to determine if the user is allowed to run an EXEC shell on the selected line(s).
reverse-access	Enables authorization to determine if the user is allowed reverse access privileges.
default	(Optional) The name of the default method list, created with the aaa authorization command.
<i>list-name</i>	(Optional) 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.

Defaults Authorization is not enabled.

Command Modes Line configuration

Command History	Release	Modification
	11.3 T	This command was introduced.

Usage Guidelines After you enable the **aaa authorization** command and 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.

Examples The following example enables command authorization (for level 15) using the method list named charlie on line 10:

```
line 10
  authorization commands 15 charlie
```

Related Commands	Command	Description
	aaa authorization	Sets parameters that restrict network access to a user.

ppp authorization

To enable AAA authorization on the selected interface, use the **ppp authorization** interface configuration command. Use the **no** form of this command to disable authorization.

ppp authorization [**default** | *list-name*]

no ppp authorization

Syntax Description

default	(Optional) The name of the method list is created with the aaa authorization command.
<i>list-name</i>	(Optional) 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.

Defaults

Authorization is disabled.

Command Modes

Interface configuration

Command History

Release	Modification
11.3 T	This command was introduced.

Usage Guidelines

After you enable the **aaa authorization** command and define a named authorization method list (or use the default method list), you must apply the defined lists to the appropriate interfaces for authorization to take place. Use the **ppp authorization** command to apply the specified method lists (or if none is specified, the default method list) to the selected interface.

Examples

The following example enables authorization on asynchronous interface 4 and uses the method list named charlie:

```
interface async 4
 encapsulation ppp
 ppp authorization charlie
```

Related Commands

Command	Description
aaa authorization	Sets parameters that restrict network access to a user.

