

# reset—switch

Use the **reset** command set to restart the system or an individual module, schedule a system reset, or cancel a scheduled reset.

**reset** [*mod* | **system** | **mindown**]

**reset** [**mindown**] **at** {*hh:mm*} [*mm/dd*] [*reason*]

**reset** [**mindown**] **in** [*hh:*] {*mm*} [*reason*]

**reset** [**cancel**]

**reset** {*nam\_mod*} [*bootdevice*[,*bootdevice*]]

Syntax	Description
<i>mod</i>	(Optional) Number of the module to be restarted.
<b>system</b>	(Optional) Keyword to reset the system.
<b>mindown</b>	(Optional) Keyword to perform a reset as part of a minimal downtime software upgrade in a system with a redundant supervisor engine.
<b>at</b>	Keyword to schedule a system reset at a specific future time.
<i>hh:mm</i>	Hour and minute of the scheduled reset.
<i>mm/dd</i>	(Optional) Month and day of the scheduled reset.
<i>reason</i>	(Optional) Reason for the reset.
<b>in</b>	Keyword to schedule a system reset in a specific time.
<i>hh</i>	(Optional) Number of hours into the future to reset the switch.
<i>mm</i>	Number of minutes into the future to reset the switch.
<b>cancel</b>	(Optional) Keyword to cancel the scheduled reset.
<i>nam_mod</i>	Number of the NAM.
<i>bootdevice</i>	(Optional) Boot device identification; for format guidelines, see the “Usage Guidelines” section.

**Defaults** This command has no default setting.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** If you do not specify a module number (either a switching module or the active supervisor engine module), the command resets the entire system.

You can use the **reset** *mod* command to switch to the standby supervisor engine, where *mod* is the module number of the active supervisor engine.

You can use the **reset mindown** command to reset the switch as part of a minimal downtime software upgrade in a system with redundant supervisor engine. For complete information on performing a minimal downtime software upgrade, refer to the *Catalyst 6000 Family Software Configuration Guide*.



### Caution

If you make configuration changes after entering the **reset mindown** command but before the active supervisor engine resets, the changes are not saved. Input from the CLI is still accepted by the switch while the standby supervisor engine is reset, but any changes you make to the configuration between the time when you enter the **reset mindown** command and the time when the supervisor engine comes online running the new software image are not saved or synchronized with the standby supervisor engine.

If you reset an intelligent module (such as the Catalyst 6000 family MSM or MSFC), both the module hardware and software are completely reset.

When entering the *bootdevice*, use the format *device[:device\_qualifier]* where:

- *device* = **pcmcia**, **hdd**, **network**
- *device\_qualifier* **hdd** = number from 1 to 99
- **pcmcia** = slot0 or slot1

### Examples

This example shows how to reset the supervisor engine on a Catalyst 6000 family switch with redundant supervisor engines:

```
Console> (enable) reset 1
This command will force a switch-over to the standby supervisor module
and disconnect your telnet session.
Do you want to continue (y/n) [n]? y
Connection closed by foreign host.
host%
```

This example shows how to reset module 4:

```
Console> (enable) reset 4
This command will reset module 4 and may disconnect your telnet session.
Do you want to continue (y/n) [n]? y
Resetting module 4...
Console> (enable)
```

This example shows how to schedule a system reset for a specific future time:

```
Console> (enable) reset at 20:00
Reset scheduled at 20:00:00, Wed Mar 15 2000.
Proceed with scheduled reset? (y/n) [n]? y
Reset scheduled for 20:00:00, Wed Mar 15 2000 (in 0 day 5 hours 40 minutes).
Console> (enable)
```

This example shows how to schedule a reset for a specific future time and include a reason for the reset:

```
Console> (enable) reset at 23:00 3/15 Software upgrade to 6.1(1).
Reset scheduled at 23:00:00, Wed Mar 15 2000.
Reset reason: Software upgrade to 6.1(1).
Proceed with scheduled reset? (y/n) [n]? y
Reset scheduled for 23:00:00, Wed Mar 15 2000 (in 0 day 8 hours 39 minutes).
Console> (enable)
```

This example shows how to schedule a reset with minimum downtime for a specific future time and include a reason for the reset:

```
Console> (enable) reset mindown at 23:00 3/15 Software upgrade to 6.1(1).  
Reset scheduled at 23:00:00, Wed Mar 15 2000.  
Reset reason: Software upgrade to 6.1(1).  
Proceed with scheduled reset? (y/n) [n]? y  
Reset mindown scheduled for 23:00:00, Wed Mar 15 2000 (in 0 day 8 hours 39 minutes).  
Console> (enable)
```

This example shows how to schedule a reset after a specified time:

```
Console> (enable) reset in 5:20 Configuration update  
Reset scheduled in 5 hours 20 minutes.  
Reset reason: Configuration update  
Proceed with scheduled reset? (y/n) [n]? y  
Reset scheduled for 19:56:01, Wed Mar 15 2000 (in 5 hours 20 minutes).  
Reset reason: Configuration update  
Console> (enable)
```

This example shows how to cancel a schedule reset:

```
Console> (enable) reset cancel  
Reset cancelled.  
Console> (enable)
```

---

**Related Commands**

**show reset**  
**commit**

# rollback

Use the **rollback** command set to clear changes made to the ACL edit buffer since its last save. The ACL is rolled back to its state at the last **commit** command.

**rollback qos acl** *acl\_name*

**rollback security acl** *acl\_name*

Syntax Description		
	<b>qos acl</b>	Keyword to specify QoS ACEs.
	<b>security acl</b>	Keywords to specify security ACEs.
	<i>acl_name</i>	Name that identifies the VACL whose ACEs are to be affected.

**Defaults** This command has no default setting.

**Command Types** Switch command.

**Command Modes** Privileged.

**Examples** This example shows how to clear the edit buffer of a specific QoS ACL:

```
Console (enable) rollback qos acl ip-8-1
Rollback for QoS ACL ip-8-1 is successful.
Console> (enable)
```

This example shows how to clear the edit buffer of a specific security ACL:

```
Console> (enable) rollback security acl IPACL1
IPACL1 editbuffer modifications cleared.
Console> (enable)
```

**Related Commands** **show qos acl info**  
**commit**

# session

Use the **session** command to open a session with a module (for example, the MSM or ATM), allowing you to use the module-specific CLI.

**session** *mod*

---

## Syntax Description

---

*mod*            Number of the module.

---



---

## Defaults

This command has no default setting.

---

## Command Types

Switch command.

---

## Command Modes

Privileged.

---

## Usage Guidelines

After you enter this command, the system responds with the Enter Password: prompt, if one is configured on the module.

To end the session, enter the **quit** command.

Use the **session** command to toggle between router and switch sessions.

For information on ATM commands, refer to the *ATM Software Configuration Guide and Command Reference for the Catalyst 5000 Family and 6000 Family Switches*.

---

## Examples

This example shows how to open a session with an MSM (module 4):

```
Console> session 4
Trying Router-4...
Connected to Router-4.
Escape character is '^]'.

Router>
```

---

## Related Commands

**switch console**  
**quit**

# set

Use the **set** command to display all of the ROM monitor variable names with their values.

**set**

---

**Syntax Description** This command has no arguments or keywords.

---

**Defaults** This command has no default setting.

---

**Command Types** ROM monitor command.

---

**Command Modes** Normal.

---

**Examples** This example shows how to display all of the ROM monitor variable names with their values:

```
rommon 2 > set  
PS1=rommon ! >  
BOOT=  
?=0
```

---

**Related Commands** **varname=**

# set accounting commands

Use the **set accounting commands** command set to enable command event accounting on the switch.

```
set accounting commands enable {config | all} [stop-only] {tacacs+}
```

```
set accounting commands disable
```

Syntax Description	enable	Keyword to enable the specified accounting method for commands.
	<b>config</b>	Keyword to permit accounting for configuration commands only.
	<b>all</b>	Keyword to permit accounting for all commands.
	<b>stop-only</b>	(Optional) Keyword to apply the accounting method at the command end.
	<b>tacacs+</b>	Keyword to specify TACACS+ accounting for commands.
	<b>disable</b>	Keyword to disable accounting for commands.

**Defaults** The default is accounting is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** You must configure the TACACS+ servers before you enable accounting.

**Examples** This example shows how to send records at the end of the event only using a TACACS+ server:

```
Console> (enable) set accounting commands enable config stop-only tacacs+
Accounting set to enable for commands-config events in stop-only mode.
Console> (enable)
```

**Related Commands**

- set accounting connect**
- set accounting exec**
- set accounting suppress**
- set accounting system**
- set accounting update**
- set tacacs server**
- show accounting**

# set accounting connect

Use the **set accounting connect** command set to enable accounting of outbound connection events on the switch.

```
set accounting connect enable {start-stop | stop-only} {tacacs+ | radius}
```

```
set accounting connect disable
```

Syntax Description	enable	Keyword to enable the specified accounting method for connection events.
	<b>start-stop</b>	Keyword to specify the accounting method applies at the start and stop of the connection event.
	<b>stop-only</b>	Keyword to specify the accounting method applies at the end of the connection event.
	<b>tacacs+</b>	Keyword to specify TACACS+ accounting for connection events.
	<b>radius</b>	Keyword to specify RADIUS accounting for connection events.
	<b>disable</b>	Keyword to disable accounting of connection events.

**Defaults** The default is accounting is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** You must configure the RADIUS or TACACS+ servers and shared secret keys before you enable accounting.

**Examples** This example shows how to enable accounting on Telnet and remote login sessions, generating records at stop only using a TACACS+ server:

```
Console> (enable) set accounting connect enable stop-only tacacs+
Accounting set to enable for connect events in stop-only mode..
Console> (enable)
```

---

**Related Commands**

**set accounting commands**  
**set accounting exec**  
**set accounting suppress**  
**set accounting system**  
**set accounting update**  
**set radius key**  
**set radius server**  
**set tacacs key**  
**set tacacs server**  
**show accounting**

# set accounting exec

Use the **set accounting exec** command set to enable accounting of normal login sessions on the switch.

```
set accounting exec enable {start-stop | stop-only} {tacacs+ | radius}
```

```
set accounting exec disable
```

Syntax Description	enable	Keyword to enable the specified accounting method for normal login sessions.
	<b>start-stop</b>	Keyword to specify the accounting method applies at the start and stop of the normal login sessions.
	<b>stop-only</b>	Keyword to specify the accounting method applies at the end of the normal login sessions.
	<b>tacacs+</b>	Keyword to specify TACACS+ accounting for normal login sessions.
	<b>radius</b>	Keyword to specify RADIUS accounting for normal login sessions.
	<b>disable</b>	Keyword to disable accounting for normal login sessions.

**Defaults** The default is accounting is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** You must configure the RADIUS or TACACS+ servers and shared secret keys before you enable accounting.

**Examples** This example shows how to enable accounting of normal login sessions, generating records at start and stop using a RADIUS server:

```
Console> (enable) set accounting exec enable start-stop radius
Accounting set to enable for exec events in start-stop mode.
Console> (enable)
```

This example shows how to enable accounting of normal login sessions, generating records at stop using a TACACS+ server:

```
Console> (enable) set accounting exec enable stop-only tacacs+
Accounting set to enable for exec events in stop-only mode.
Console> (enable)
```

---

**Related Commands**

**set accounting commands**  
**set accounting connect**  
**set accounting suppress**  
**set accounting system**  
**set accounting update**  
**set radius key**  
**set radius server**  
**set tacacs key**  
**set tacacs server**  
**show accounting**

# set accounting suppress

Use the **set accounting suppress** command to enable or disable suppression of accounting information for a user who has logged in without a username.

**set accounting suppress null-username {enable | disable}**

Syntax Description	Parameter	Description
	<b>null-username</b>	Keyword to specify users must have a user ID.
	<b>enable</b>	Keyword to enable suppression for a specified user.
	<b>disable</b>	Keyword to disable suppression for a specified user.

**Defaults** The default is accounting is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** You must configure the TACACS+ servers before you enable accounting.

**Examples** This example shows how to suppress accounting information for users without a username:

```
Console> (enable) set accounting suppress null-username enable
Accounting will be suppressed for user with no username.
Console> (enable)
```

This example shows how to include users without the usernames' accounting event information:

```
Console> (enable) set accounting suppress null-username disable
Accounting will be not be suppressed for user with no username.
Console> (enable)
```

**Related Commands**

- set accounting commands
- set accounting connect
- set accounting exec
- set accounting system
- set accounting update
- set tacacs server
- show accounting

# set accounting system

Use the **set accounting system** command set to enable accounting of system events on the switch.

```
set accounting system enable {start-stop | stop-only} {tacacs+ | radius}
```

```
set accounting system disable
```

Syntax Description	enable	Keyword to enable the specified accounting method for system events.
	<b>start-stop</b>	Keyword to specify the accounting method applies at the start and stop of the system event.
	<b>stop-only</b>	Keyword to specify the accounting method applies at the end of the system event.
	<b>tacacs+</b>	Keyword to specify TACACS+ accounting for system events.
	<b>radius</b>	Keyword to specify RADIUS accounting for system events.
	<b>disable</b>	Keyword to disable accounting for system events.

**Defaults** The default is accounting is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** You must configure the RADIUS or TACACS+ servers and shared secret keys before you enable accounting.

**Examples** This example shows how to enable accounting for system events, sending records only at the end of the event using a RADIUS server:

```
Console> (enable) set accounting system enable stop-only radius
Accounting set to enable for system events in start-stop mode..
Console> (enable)
```

This example shows how to enable accounting for system events, sending records only at the end of the event using a TACACS+ server:

```
Console> (enable) set accounting system enable stop-only tacacs+
Accounting set to enable for system events in start-stop mode..
Console> (enable)
```

■ set accounting system

---

**Related Commands**

**set accounting commands**  
**set accounting connect**  
**set accounting exec**  
**set accounting suppress**  
**set accounting update**  
**set radius key**  
**set radius server**  
**set tacacs key**  
**set tacacs server**  
**show accounting**

# set accounting update

Use the **set accounting update** command to configure the frequency of accounting updates.

```
set accounting update {new-info | {periodic [interval]}}
```

Syntax Description	
<b>new-info</b>	Keyword to specify update when new information is available.
<b>periodic</b>	Keyword to update on a periodic basis.
<i>interval</i>	(Optional) Periodic update interval time; valid values are from 1 to 71582 minutes.

**Defaults** The default is accounting is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** You must configure the TACACS+ servers before you enable accounting.

**Examples** This example shows how to send accounting updates every 200 minutes:

```
Console> (enable) set accounting update periodic 200
Accounting updates will be periodic at 200 minute intervals.
Console> (enable)
```

This example shows how to send accounting updates only when there is new information:

```
Console> (enable) set accounting update new-info
Accounting updates will be sent on new information only.
Console> (enable)
```

**Related Commands**

- set accounting commands**
- set accounting connect**
- set accounting exec**
- set accounting suppress**
- set accounting system**
- set tacacs server**
- show accounting**

# set alias

Use the **set alias** command to define aliases (shorthand versions) of commands.

```
set alias name command [parameter] [parameter]
```

Syntax Description	
<i>name</i>	Alias being created.
<i>command</i>	Command for which the alias is being created.
<i>parameter</i>	(Optional) Parameters that apply to the command for which an alias is being created.

**Defaults** The default is no aliases are configured.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** The name **all** cannot be defined as an alias. Reserved words cannot be defined as aliases. For additional information about *parameter*, see the specific command for information about applicable parameters.

**Examples** This example shows how to set the alias for the **clear arp** command as arpdel:

```
Console> (enable) set alias arpdel clear arp
Command alias added.
Console> (enable)
```

**Related Commands** **clear alias**  
**show alias**

# set arp

Use the **set arp** command set to add IP address-to-MAC address mapping entries to the ARP table and to set the ARP aging time for the table.

```
set arp [dynamic | permanent | static] {ip_addr hw_addr}
```

```
set arp agingtime agingtime
```

Syntax Description	
<b>dynamic</b>	(Optional) Keyword to specify that entries are subject to ARP aging updates.
<b>permanent</b>	(Optional) Keyword to specify that permanent entries are stored in NVRAM until they are removed by the <b>clear arp</b> or <b>clear config</b> command.
<b>static</b>	(Optional) Keyword to specify that entries are not subject to ARP aging updates.
<i>ip_addr</i>	IP address or IP alias to map to the specified MAC address.
<i>hw_addr</i>	MAC address to map to the specified IP address or IP alias.
<b>agingtime</b>	Keyword to set the period of time after which an ARP entry is removed from the ARP table.
<i>agingtime</i>	Number of seconds that entries will remain in the ARP table before being deleted; valid values are from 0 to 1,000,000 seconds. Setting this value to 0 disables aging.

**Defaults** The default is no ARP table entries exist; ARP aging is set to 1200 seconds.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** When entering the *hw\_addr*, use a 6-hexadecimal byte MAC address in canonical (00-11-22-33-44-55) or noncanonical (00:11:22:33:44:55) format.

Static (nonpermanent) entries remain in the ARP table until you reset the active supervisor engine.

**Examples** This example shows how to configure a dynamic ARP entry mapping that will age out after the configured ARP aging time:

```
Console> (enable) set arp dynamic 198.133.219.232 00-00-0c-40-0f-bc
ARP entry added.
Console> (enable)
```

This example shows how to set the aging time for the ARP table to 1800 seconds:

```
Console> (enable) set arp agingtime 1800  
ARP aging time set to 1800 seconds.  
Console> (enable)
```

This example shows how to configure a permanent ARP entry, which will remain in the ARP cache after a system reset:

```
Console> (enable) set arp permanent 198.146.232.23 00-00-0c-30-0f-bc  
Permanent ARP entry added as  
198.146.232.23 at 00-00-0c-30-0f-bc on vlan 5  
Console> (enable)
```

This example shows how to configure a static ARP entry, which will be removed from the ARP cache after a system reset:

```
Console> (enable) set arp static 198.144.239.22 00-00-0c-50-0f-bc  
Static ARP entry added as  
198.144.239.22 at 00-00-0c-50-0f-bc on vlan 5  
Console> (enable)
```

---

#### Related Commands

**clear arp**  
**show arp**

# set authentication enable

Use the **set authentication enable** command set to enable authentication using the TACACS+, RADIUS, or Kerberos server to determine if you have privileged access permission.

```
set authentication enable tacacs {enable | disable} [all | console | http] [telnet] [primary]
```

```
set authentication enable radius {enable | disable} [all | console | http] [telnet] [primary]
```

```
set authentication enable local {enable | disable} [all | console | http] [telnet] [primary]
```

```
set authentication enable kerberos {enable | disable} [all | console | http] [telnet] [primary]
```

Syntax Description	
<b>tacacs</b>	Keyword to specify TACACS+ authentication for login.
<b>enable</b>	Keyword to enable the specified authentication method for login.
<b>disable</b>	Keyword to disable the specified authentication method for login.
<b>all</b>	(Optional) Keyword to apply the authentication method to all session types.
<b>console</b>	(Optional) Keyword to specify the authentication method for console sessions.
<b>http</b>	(Optional) Keyword to specify the specified authentication method HTTP sessions.
<b>telnet</b>	(Optional) Keyword to specify the authentication method for Telnet sessions.
<b>primary</b>	(Optional) Keyword to specify the specified authentication method be tried first.
<b>radius</b>	Keyword to specify RADIUS authentication for login.
<b>local</b>	Keyword to specify local authentication for login.
<b>kerberos</b>	Keyword to specify Kerberos authentication for login.

**Defaults** The default is local authentication is enabled for console and Telnet sessions. RADIUS, TACACS+, and Kerberos are disabled for all session types.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** Use authentication configuration for both console and Telnet connection attempts unless you use the **console** and **telnet** keywords to specify the authentication methods for each connection type individually.

---

**Examples**

This example shows how to use the TACACS+ server to determine if a user has privileged access permission:

```
Console> (enable) set authentication enable tacacs enable  
tacacs enable authentication set to enable for console, telnet and http session.  
Console> (enable)
```

This example shows how to use the local password to determine if the user has privileged access permission:

```
Console> (enable) set authentication enable local enable  
local enable authentication set to enable for console, telnet and http session.  
Console> (enable)
```

This example shows how to use the RADIUS server to determine if a user has privileged access permission for all session types:

```
Console> (enable) set authentication enable radius enable  
radius enable authentication set to enable for console, telnet and http session.  
Console> (enable)
```

This example shows how to use the TACACS+ server to determine if a user has privileged access permission for all session types:

```
Console> (enable) set authentication enable tacacs enable console  
tacacs enable authentication set to enable for console session.  
Console> (enable)
```

This example shows how to set the Kerberos server to be used first:

```
Console> (enable) set authentication enable kerberos enable primary  
kerberos enable authentication set to enable for console, telnet and http session  
n as primary authentication method.  
Console> (enable)
```

---

**Related Commands**

**set authentication login**  
**show authentication**

# set authentication login

Use the **set authentication login** command set to enable TACACS+, RADIUS, or Kerberos as the authentication method for login.

```
set authentication login local {enable | disable} [all | console | http | telnet]
```

```
set authentication login tacacs {enable | disable} [all | console | http | telnet] [primary]
```

```
set authentication login radius {enable | disable} [all | console | http | telnet] [primary]
```

```
set authentication login kerberos {enable | disable} [all | console | http | telnet] [primary]
```

Syntax Description		
<b>local</b>	Keyword to specify local password to determine if you have access permission to the switch.	
<b>enable</b>	Keyword to enable the specified authentication method for login.	
<b>disable</b>	Keyword to disable the specified authentication method for login.	
<b>all</b>	(Optional) Keyword to specify the authentication method for all session types.	
<b>console</b>	(Optional) Keyword to specify the authentication method for console sessions.	
<b>http</b>	(Optional) Keyword to specify the authentication method for HTTP sessions or to set HTTP sessions as the primary authentication method.	
<b>telnet</b>	(Optional) Keyword to specify the authentication method for Telnet sessions.	
<b>tacacs</b>	Keyword to specify the use of the TACACS+ server password to determine if you have access permission to the switch.	
<b>radius</b>	Keyword to specify the use of the RADIUS server password to determine if you have access permission to the switch.	
<b>kerberos</b>	Keyword to specify the Kerberos server password to determine if you have access permission to the switch.	

**Defaults** The default is local authentication is the primary authentication method for login.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** This command allows you to choose the authentication method for the web interface. If you configure the authentication method for the HTTP session as RADIUS, then the username or password is validated using the RADIUS protocol and TACACS+ and Kerberos authentication is set to disable for the HTTP sessions. By default, the HTTP login is validated using the local login password.

You can specify that the authentication method for **console**, **telnet**, **http**, or **all** by entering the **console**, **telnet**, **http**, or **all** keywords. If you do not specify **console**, **telnet**, **http**, or **all**, the authentication method default is for **all** sessions.

---

**Examples**

This example shows how to disable TACACS+ authentication access for Telnet sessions:

```
Console> (enable) set authentication login tacacs disable telnet  
tacacs login authentication set to disable for the telnet sessions.  
Console> (enable)
```

This example shows how to disable RADIUS authentication access for console sessions:

```
Console> (enable) set authentication login radius disable console  
radius login authentication set to disable for the console sessions.  
Console> (enable)
```

This example shows how to disable Kerberos authentication access for Telnet sessions:

```
Console> (enable) set authentication login kerberos disable telnet  
kerberos login authentication set to disable for the telnet sessions.  
Console> (enable)
```

This example shows how to set TACACS+ authentication access as the primary method for HTTP sessions:

```
Console> (enable) set authentication login tacacs enable http primary  
tacacs login authentication set to enable for HTTP sessions as primary authentication  
method.  
Console> (enable)
```

---

**Related Commands**

**set authentication enable**  
**show authentication**

# set authorization commands

Use the **set authorization commands** command set to enable authorization of command events on the switch.

```
set authorization commands enable {config | all} {option} {fallbackoption}
[console | telnet | both]
```

```
set authorization commands disable [console | telnet | both]
```

Syntax Description		
<b>enable</b>	Keyword to enable the specified authorization method for commands.	
<b>config</b>	Keyword to permit authorization for configuration commands only.	
<b>all</b>	Keyword to permit authorization for all commands.	
<i>option</i>	Switch response to an authorization request; valid values are <b>tacacs+</b> , <b>if-authenticated</b> , and <b>none</b> . See the “Usage Guidelines” section for valid value definitions.	
<i>fallbackoption</i>	Switch fallback response to an authorization request if the TACACS+ server is down or not responding; valid values are <b>tacacs+</b> , <b>deny</b> , <b>if-authenticated</b> , and <b>none</b> . See the “Usage Guidelines” section for valid value definitions.	
<b>console</b>	(Optional) Keyword to specify the authorization method for console sessions.	
<b>telnet</b>	(Optional) Keyword to specify the authorization method for Telnet sessions.	
<b>both</b>	(Optional) Keyword to specify the authorization method for both console and Telnet sessions.	
<b>disable</b>	Keyword to disable authorization of command events.	

**Defaults** The default is authorization is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** When you define the *option* and *fallbackoption* values:

- **tacacs+** specifies the TACACS+ authorization method.
- **deny** does not let you proceed.
- **if-authenticated** allows you to proceed with your action if you have been authenticated.
- **none** allows you to proceed without further authorization in case the TACACS+ server does not respond.

---

**Examples**

This example shows how to enable authorization for all commands with the **if-authenticated** option and **none fallbackoption**:

```
Console> (enable) set authorization commands enable all if-authenticated none  
Successfully enabled commands authorization.  
Console> (enable)
```

This example shows how to disable command authorization:

```
Console> (enable) set authorization commands disable  
Successfully disabled commands authorization.  
Console> (enable)
```

---

**Related Commands**

**set authorization enable**  
**set authorization exec**  
**show authorization**

# set authorization enable

Use the **set authorization enable** command set to enable authorization of privileged mode sessions on the switch.

```
set authorization enable enable {option} {fallbackoption} [console | telnet | both]
```

```
set authorization enable disable [console | telnet | both]
```

Syntax Description		
<b>enable</b>		Keyword to enable the specified authorization method.
<i>option</i>		Switch response to an authorization request; valid values are <b>tacacs+</b> , <b>if-authenticated</b> , and <b>none</b> . See the “Usage Guidelines” section for valid value definitions.
<i>fallbackoption</i>		Switch fallback response to an authorization request if the TACACS+ server is down or not responding; valid values are <b>tacacs+</b> , <b>deny</b> , <b>if-authenticated</b> , and <b>none</b> . See the “Usage Guidelines” section for valid value definitions.
<b>console</b>		(Optional) Keyword to specify the authorization method for console sessions.
<b>telnet</b>		(Optional) Keyword to specify the authorization method for Telnet sessions.
<b>both</b>		(Optional) Keyword to specify the authorization method for both console and Telnet sessions.
<b>disable</b>		Keyword to disable the authorization method.

**Defaults** The default is authorization is disabled.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** When you define the *option* and *fallbackoption* values:

- **tacacs+** specifies the TACACS+ authorization method.
- **deny** does not let you proceed.
- **if-authenticated** allows you to proceed with your action if you have authentication.
- **none** allows you to proceed without further authorization in case the TACACS+ server does not respond.

---

**Examples**

This example shows how to enable authorization of configuration commands in enable, privileged login mode, sessions:

```
Console> (enable) set authorization enable enable if-authenticated none  
Successfully enabled enable authorization.  
Console> (enable)
```

This example shows how to disable enable mode authorization:

```
Console> (enable) set authorization enable disable  
Successfully disabled enable authorization.  
Console> (enable)
```

---

**Related Commands**

**set authorization commands**  
**set authorization exec**  
**show authorization**

# set authorization exec

Use the **set authorization exec** command set to enable authorization of exec, normal login mode, session events on the switch.

```
set authorization exec enable {option} {fallbackoption} [console | telnet | both]
```

```
set authorization exec disable [console | telnet | both]
```

Syntax Description		
<b>enable</b>		Keyword to enable the specified authorization method.
<i>option</i>		Switch response to an authorization request; valid values are <b>tacacs+</b> , <b>if-authenticated</b> , and <b>none</b> . See the “Usage Guidelines” section for valid value definitions.
<i>fallbackoption</i>		Switch fallback response to an authorization request if the TACACS+ server is down or not responding; valid values are <b>tacacs+</b> , <b>deny</b> , <b>if-authenticated</b> , and <b>none</b> . See the “Usage Guidelines” section for valid value definitions.
<b>console</b>		(Optional) Keyword to specify the authorization method for console sessions.
<b>telnet</b>		(Optional) Keyword to specify the authorization method for Telnet sessions.
<b>both</b>		(Optional) Keyword to specify the authorization method for both console and Telnet sessions.
<b>disable</b>		Keyword to disable authorization method.

**Defaults** The default is authorization is denied.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** When you define the *option* and *fallbackoption* values:

- **tacacs+** specifies the TACACS+ authorization method.
- **deny** fails authorization if the TACACS+ server does not respond.
- **if-authenticated** allows you to proceed with your action if the TACACS+ server does not respond and you have authentication.
- **none** allows you to proceed without further authorization if the TACACS+ server does not respond.

---

**Examples**

This example shows how to enable authorization of configuration commands in exec, normal login mode, sessions:

```
Console> (enable) set authorization exec enable if-authenticated none  
Successfully enabled exec authorization.  
Console> (enable)
```

This example shows how to disable exec mode authorization:

```
Console> (enable) set authorization exec disable  
Successfully disabled exec authorization.  
Console> (enable)
```

---

**Related Commands**

**set authorization commands**  
**set authorization enable**  
**show authorization**

# set banner motd

Use the **set banner motd** command to program an MOTD banner to appear before session login.

```
set banner motd c [text] c
```

## Syntax Description

<i>c</i>	Delimiting character used to begin and end the message.
<i>text</i>	(Optional) Message of the day.

## Defaults

This command has no default setting.

## Command Types

Switch command.

## Command Modes

Privileged.

## Usage Guidelines

The banner may contain no more than 3,070 characters, including tabs. Tabs display as eight characters but take only one character of memory.

You can use either the **clear banner motd** command or the **set banner motd cc** command to clear the message-of-the-day banner.

## Examples

This example shows how to set the message of the day using the pound sign (#) as the delimiting character:

```
Console> (enable) set banner motd #
** System upgrade at 6:00am Tuesday.
** Please log out before leaving on Monday. #
MOTD banner set.
Console> (enable)
```

This example shows how to clear the message of the day:

```
Console> (enable) set banner motd ##
MOTD banner cleared.
Console> (enable)
```

## Related Commands

**clear banner motd**

## set boot auto-config

Use the **set boot auto-config** command to specify one or more configuration files to use to configure the switch at bootup. The list of configuration files is stored in the CONFIG\_FILE environment variable.

```
set boot auto-config device:filename [;device:filename...] [mod]
```

Syntax Description	
<i>device:</i>	Device where the startup configuration file resides.
<i>filename</i>	Name of the startup configuration file.
<i>mod</i>	(Optional) Module number of the supervisor engine containing the Flash device.

**Defaults** The default CONFIG\_FILE is slot0:switch.cfg.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** The **set boot auto-config** command always overwrites the existing CONFIG\_FILE environment variable settings (you cannot prepend or append a file to the variable contents).

If you specify multiple configuration files, you must separate the files with a semicolon (;).

To set the recurrence on other supervisor engines and switches, use the **set boot config-register auto-config** command.

**Examples** This example shows how to specify a single configuration file environment variable:

```
Console> (enable) set boot auto-config slot0:cfgfile2
CONFIG_FILE variable = slot0:cfgfile2
WARNING: nvram configuration may be lost during next bootup,
        and re-configured using the file(s) specified.
Console> (enable)
```

This example shows how to specify multiple configuration file environment variables:

```
Console> (enable) set boot auto-config slot0:cfgfile1;slot0:cfgfile2
CONFIG_FILE variable = slot0:cfgfile1;slot0:cfgfile2
WARNING: nvram configuration may be lost during next bootup,
        and re-configured using the file(s) specified.
Console> (enable)
```

---

**Related Commands**

- set boot config-register**
- set boot system flash**
- show boot**

# set boot config-register

Use the **set boot config-register** command set to configure the boot configuration register value.

**set boot config-register 0xvalue** [*mod*]

**set boot config-register baud** {1200 | 2400 | 4800 | 9600} [*mod*]

**set boot config-register ignore-config** {enable | disable} [*mod*]

**set boot config-register boot** {rommon | bootflash | system} [*mod*]

Syntax Description		
<b>0xvalue</b>	Keyword to set the 16-bit configuration register value.	
<i>mod</i>	(Optional) Module number of the supervisor engine containing the Flash device.	
<b>baud 1200   2400   4800   9600</b>	Keywords to specify the console baud rate.	
<b>ignore-config</b>	Keywords to set the ignore-config feature.	
<b>enable</b>	Keyword to enable the specified feature.	
<b>disable</b>	Keyword to disable the specified feature.	
<b>boot</b>	Keyword to specify the boot image to use on the next restart.	
<b>rommon</b>	Keyword to specify booting from the ROM monitor.	
<b>bootflash</b>	Keyword to specify booting from the bootflash.	
<b>system</b>	Keyword to specify booting from the system.	

## Defaults

The defaults are as follows:

- Configuration register value is 0x10F, which causes the switch to boot from what is specified by the BOOT environment variable.
- Baud rate is set to 9600.
- **ignore-config** parameter is disabled.

## Command Types

Switch command.

## Command Modes

Privileged.

## Usage Guidelines

We recommend that you use only the **rommon** and **system** options to the **set boot config-register boot** command.

Each time you enter one of the **set boot config-register** commands, the system displays all current configuration-register information (the equivalent of entering the **show boot** command).

The baud rate specified in the configuration register is used by the ROM monitor only and is different from the baud rate specified by the **set system baud** command.

When you enable the **ignore-config** feature, the system software ignores the configuration. Enabling the **ignore-config** parameter is the same as entering the **clear config all** command; that is, it clears the entire configuration stored in NVRAM the next time the switch is restarted.

---

**Examples**

This example shows how to specify booting from the ROM monitor:

```
Console> (enable) set boot config-register boot rommon
Configuration register is 0x100
ignore-config: disabled
console baud: 9600
boot: the ROM monitor
Console> (enable)
```

This example shows how to specify the default 16-bit configuration register value:

```
Console> (enable) set boot config-register 0x12f
Configuration register is 0x12f
break: disabled
ignore-config: disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to change the ROM monitor baud rate to 4800:

```
Console> (enable) set boot config-register baud 4800
Configuration register is 0x90f
ignore-config: disabled
console baud: 4800
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to ignore the configuration information stored in NVRAM the next time the switch is restarted:

```
Console> (enable) set boot config-register ignore-config enable
Configuration register is 0x94f
ignore-config: enabled
console baud: 4800
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to specify rommon as the boot image to use on the next restart:

```
Console> (enable) set boot config-register boot rommon
Configuration register is 0x100
ignore-config: disabled
console baud: 9600
boot: the ROM monitor
Console> (enable)
```

---

**Related Commands**

**set config acl**  
**set boot auto-config**  
**set boot system flash**  
**show boot**  
**copy**  
**show config**

# set boot config-register auto-config

Use the **set boot config-register auto-config** command set to configure auto-config file dispensation.

**set boot config-register auto-config** { **recurring** | **non-recurring** } [*mod*]

**set boot config-register auto-config** { **overwrite** | **append** }

**set boot config-register auto-config sync** { **enable** | **disable** }

Syntax Description		
<b>recurring</b>	Keyword to set auto-config to recurring and specify the switch retains the contents of the CONFIG_FILE environment variable after the switch is reset or power cycled and configured.	
<b>non-recurring</b>	Keyword to set auto-config to nonrecurring and cause the switch to clear the contents of the CONFIG_FILE environment variable after the switch is reset or power cycled and before the switch is configured.	
<i>mod</i>	(Optional) Module number of the supervisor engine containing the Flash device.	
<b>overwrite</b>	Keyword to cause the auto-config file to overwrite the NVRAM configuration.	
<b>append</b>	Keyword to cause the auto-config file to append to the file currently in the NVRAM configuration.	
<b>sync enable</b>   <b>disable</b>	Keywords to enable or disable synchronization of the auto-config file.	

## Defaults

The defaults are as follows:

- **overwrite**
- **non-recurring**
- **sync is disable**

## Command Types

Switch command.

## Command Modes

Privileged.

## Usage Guidelines

The **auto-config overwrite** command clears the NVRAM configuration before executing the Flash configuration file. The **auto-config append** command executes the Flash configuration file before clearing the NVRAM configuration.

If you delete the auto-config Flash file(s) on the supervisor engine, the files will also be deleted on the standby supervisor engine.

If you enable synchronization, the CONFIG\_FILE variable from the active file is made identical on the standby supervisor engine. Each auto-config file on the active supervisor engine is compared against each corresponding auto-config file on the standby supervisor engine. Two files are considered identical if the 'CRC' is the same. If a file on the standby and active supervisor engine is not identical, a new file is generated on the standby supervisor engine. If a file already exists on the standby supervisor engine, it is overwritten with the file from the active supervisor engine.

If you use the **set boot auto-config bootflash:switch.cfg** with the overwrite option, you must use the **copy config bootflash:switch.cfg** command to save the switch configuration to the auto-config file.

If you use the **set boot auto-config bootflash:switchapp.cfg** with the append option, you can use the **copy acl config bootflash:switchapp.cfg** command to save the switch configuration to the auto-config file.

If the ACL configuration location is set to Flash memory, the following message is displayed after every commit operation for either security or QoS. Use the **copy** commands to save your ACL configuration to Flash memory. If you reset the system and you made one or more commits but did not copy commands to one of the files specified in the CONFIG\_FILE variable, the following message displays:

```
Warning: System ACL configuration has been modified but not saved to Flash.
```

The files used with the **recurring** and **non-recurring** options are those specified by the CONFIG\_FILE environment variable.

## Examples

This example shows how to specify the ACL configuration Flash file at system startup:

```
Console> (enable) set boot auto-config bootflash:switchapp.cfg
Console> (enable) set boot config-register auto-config recurring
Console> (enable)
```

This example shows how to ignore the configuration information stored in NVRAM the next time the switch is restarted:

```
Console> (enable) set boot config-register auto-config non-recurring
Configuration register is 0x2102
ignore-config: disabled
auto-config: non-recurring, overwrite, auto-sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to append the auto-config file to the file currently in the NVRAM configuration:

```
Console> (enable) set boot config-register auto-config append
Configuration register is 0x2102
ignore-config: disabled
auto-config: non-recurring, append, auto-sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to use the auto-config overwrite option to save the ACL configuration to a bootflash file:

```
Console> (enable) copy config bootflash: switch.cfg
Console> (enable) set boot auto-config bootflash:switch.cfg
Console> (enable) set boot config-register auto-config overwrite
Console> (enable)
```

**Caution**

---

The following two examples assume that you have saved the ACL configuration to the bootflash:switchapp.cfg file.

---

This example shows how to enable synchronization of the auto-config file:

```
Console> (enable) set boot config-register auto-config sync enable
Configuration register is 0x2102
ignore-config: disabled
auto-config: non-recurring, append, auto-sync enabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to disable synchronization of the auto-config file:

```
Console> (enable) set boot config-register auto-config sync disable
Configuration register is 0x2102
ignore-config: disabled
auto-config: non-recurring, append, auto-sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

---

**Related Commands**

**set boot config-register**  
**set boot system flash**  
**show boot**

# set boot device

Use the **set boot device** command to set the NAM boot environment.

**set boot device** *bootseq*[,*bootseq*] *mod*

Syntax Description		
	<i>bootseq</i>	Device where the startup configuration file resides; see the “Usage Guidelines” section for format guidelines. The second <i>bootseq</i> is optional.
	<i>mod</i>	Number of the module containing the Flash device.

**Defaults** This command has no default setting.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** When you enter the **set boot device** command, the existing boot string in the supervisor engine NVRAM is always overwritten.

When you enter the *bootseq*, use the following guidelines:

- *bootseq* = *bootdevice*[:*bootdevice-qualifier*]
- *bootdevice* is the device where the startup configuration file resides; valid values are **pcmcia**, **hdd**, or **network**.
- *bootdevice-qualifier* is the name of the startup configuration file; valid values for **hdd** are from 1 to 99, and for **pcmcia**, valid values are slot0 or slot1.
- The colon between *bootdevice* and *bootdevice-qualifier* is required.
- You can enter multiple *bootseq* by separating each entry with a comma; 15 is the maximum number of boot sequences you can enter.

The supervisor engine does not validate the boot device you specify, but simply stores the boot device list in NVRAM.

This command is supported by the NAM module only.

**Examples** This example shows how to specify the boot environment to boot to the maintenance partition of the NAM on module 2:

```
Console> (enable) set boot device hdd:2 2
Device BOOT variable = hdd:2
Warning: Device list is not verified but still set in the boot string.
Console> (enable)
```

This example shows how to specify multiple boot environments on module 5:

```
Console> (enable) set boot device hdd,hdd:5,pcmcia:slot0,network,hdd:6 5
Device BOOT variable = hdd,hdd:5,pcmcia:slot0,network,hdd:6
Warning:Device list is not verified but still set in the boot string.
Console> (enable)
```

---

**Related Commands**

**clear boot device**  
**show boot device**

# set boot system flash

Use the **set boot system flash** command to set the BOOT environment variable that specifies a list of images the switch loads at startup.

```
set boot system flash device:[filename] [prepend] [mod]
```

<b>Syntax Description</b>	<i>device</i> : Device where the Flash resides.
	<i>filename</i> (Optional) Name of the configuration file.
	<b>prepend</b> (Optional) Keyword to place the device first in the list of boot devices.
	<i>mod</i> (Optional) Module number of the supervisor engine containing the Flash device.

**Defaults** This command has no default setting.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** A colon (:) is required after the specified device.

You can enter several **boot system** commands to provide a fail-safe method for booting the switch. The system stores and executes the **boot system** commands in the order in which you enter them.

Remember to clear the old entry when building a new image with a different filename in order to use the new image.

If the file does not exist (for example, if you entered the wrong filename), then the filename is appended to the bootstring, and this message displays, “Warning: File not found but still added in the bootstring.”

If the file does exist, but is not a supervisor engine image, the file is not added to the bootstring, and this message displays, “Warning: file found but it is not a valid boot image.”

**Examples** This example shows how to append the filename cat6000-sup.5-5-1.bin on device bootflash to the BOOT environment variable:

```
Console> (enable) set boot system flash bootflash:cat6000-sup.5-5-1.bin
BOOT variable = bootflash:cat6000-sup.5-4-1.bin,1;bootflash:cat6000-sup.5-5-1.bin,1;
Console> (enable)
```

This example shows how to prepend cat6000-sup.5-5-1.bin to the beginning of the boot string:

```
Console> (enable) set boot system flash bootflash:cat6000-sup.5-5-1.bin prepend
BOOT variable = bootflash:cat6000-sup.5-5-1.bin,1;bootflash:cat6000-sup.5-4-1.bin,1;
Console> (enable)
```

■ set boot system flash

---

**Related Commands**    **clear boot system**  
                              **show boot**

# set cam

Use the **set cam** command set to add entries into the CAM table and set the aging time for the CAM table.

```
set cam { dynamic | static | permanent } { unicast_mac | route_descr } mod/port [vlan]
```

```
set cam { static | permanent } { multicast_mac } mod/ports.. [vlan]
```

```
set cam agingtime vlan agingtime
```

Syntax Description		
<b>dynamic</b>	Keyword to specify that entries are subject to aging.	
<b>static</b>	Keyword to specify that entries are not subject to aging.	
<b>permanent</b>	Keyword to specify that permanent entries are stored in NVRAM until they are removed by the <b>clear cam</b> or <b>clear config</b> command.	
<i>unicast_mac</i>	MAC address of the destination host used for a unicast.	
<i>route_descr</i>	Route descriptor of the “next hop” relative to this switch; valid values are from 0 to 0xffff.	
<i>mod/port</i>	Number of the module and the port on the module.	
<i>vlan</i>	(Optional) Number of the VLAN.	
<i>multicast_mac</i>	MAC address of the destination host used for a multicast.	
<i>mod/ports..</i>	Number of the module and the ports on the module.	
<b>agingtime</b>	Keyword to set the period of time after which an entry is removed from the table.	
<i>agingtime</i>	Number of seconds (0 to 1,000,000) that dynamic entries remain in the table before being deleted. Setting the aging time to 0 disables aging.	

**Defaults** The default configuration has a local MAC address, spanning tree address (01-80-c2-00-00-00), and CDP multicast address for destination port 1/3 (the NMP). The default aging time for all configured VLANs is 300 seconds.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** If the given MAC address is a multicast address (the least significant bit of the most significant byte is set to 1) or broadcast address (ff-ff-ff-ff-ff-ff) and you specify multiple ports, the ports must all be in the same VLAN. If the given address is a unicast address and you specify multiple ports, the ports must be in different VLANs.

The **set cam** command does not support the MSM.

If you enter a route descriptor with no VLAN parameter specified, the default is the VLAN already associated with the port. If you enter a route descriptor, you may only use a single port number (of the associated port).

The *vlan* number is optional unless you are setting CAM entries to dynamic, static, or permanent for a trunk port, or if you are using the **agingtime** keyword.

If port(s) are trunk ports, you must specify the VLAN.

Static (nonpermanent) entries will remain in the table until the active supervisor engine is reset.

The *route\_descr* variable is entered as two hexadecimal bytes in the following format: 004F. Do not use a “-” to separate the bytes.

---

## Examples

This example shows how to set the CAM table aging time to 300 seconds:

```
Console> (enable) set cam agingtime 1 300
Vlan 1 CAM aging time set to 300 seconds.
Console> (enable)
```

This example shows how to add a unicast entry to the table for module 2, port 9:

```
Console> (enable) set cam static 00-00-0c-a0-03-fa 2/9
Static unicast entry added to CAM table.
Console> (enable)
```

This example shows how to add a permanent multicast entry to the table for module 1, port 1, and module 2, ports 1, 3, and 8 through 12:

```
Console> (enable) set cam permanent 01-40-0b-a0-03-fa 1/1,2/1,2/3,2/8-12
Permanent multicast entry added to CAM table.
Console> (enable)
```

---

## Related Commands

**clear cam**  
**show cam**

# set cdp

Use the **set cdp** command set to enable, disable, or configure CDP features globally on all ports or on specified ports.

**set cdp** {**enable** | **disable**} {*mod/ports...*}

**set cdp interval** *interval*

**set cdp holdtime** *holdtime*

**set cdp version** **v1** | **v2**

Syntax Description		
<b>enable</b>	Keyword to enable the CDP feature.	
<b>disable</b>	Keyword to disable the CDP feature.	
<i>mod/ports..</i>	Number of the module and the ports on the module.	
<b>interval</b>	Keyword to specify the CDP message interval value.	
<i>interval</i>	Number of seconds the system waits before sending a message; valid values are from 5 to 900 seconds.	
<b>holdtime</b>	Keyword to specify the global Time-To-Live value.	
<i>holdtime</i>	Number of seconds for the global Time-To-Live value; valid values are from 10 to 255 seconds.	
<b>version</b> <b>v1</b>   <b>v2</b>	Keywords to specify the CDP version number.	

**Defaults** The default system configuration has CDP enabled. The message interval is set to 60 seconds for every port; the default Time-To-Live value has the message interval globally set to 180 seconds. The default CDP version is version 2.

**Command Types** Switch command.

**Command Modes** Privileged.

**Usage Guidelines** The **set cdp version** command allows you to globally set the highest version number of CDP packets to send.

If you enter the global **set cdp enable** or **disable** command, CDP is globally configured. If CDP is globally disabled, CDP is automatically disabled on all interfaces, but the per-port **enable** (or **disable**) configuration is not changed. If CDP is globally enabled, whether CDP is running on an interface or not depends on its per-port configuration.

If you configure CDP on a per-port basis, you can enter the *mod/port* as a single module and port or a range of ports; for example, 2/1-12,3/5-12.

---

**Examples**

This example shows how to enable the CDP message display for port 1 on module 2:

```
Console> (enable) set cdp enable 2/1  
CDP enabled on port 2/1.  
Console> (enable)
```

This example shows how to disable the CDP message display for port 1 on module 2:

```
Console> (enable) set cdp disable 2/1  
CDP disabled on port 2/1.  
Console> (enable)
```

This example shows how to specify the CDP message interval value:

```
Console> (enable) set cdp interval 400  
CDP interval set to 400 seconds.  
Console> (enable)
```

This example shows how to specify the global Time-To-Live value:

```
Console> (enable) set cdp holdtime 200  
CDP holdtime set to 200 seconds.  
Console> (enable)
```

---

**Related Commands**

**show cdp**

# set channel cost

Use the **set channel cost** command to set the channel path cost and adjust the port costs of the ports in the channel automatically.

```
set channel cost channel_id | all [cost]
```

## Syntax Description

<i>channel_id</i>	Number of the channel identification.
<b>all</b>	Keyword to configure all channels.
<i>cost</i>	(Optional) Port costs of the ports in the channel.

## Defaults

The default is the port cost is updated automatically based on the current port costs.

## Command Types

Switch command.

## Command Modes

Privileged.

## Usage Guidelines

When you do not enter the *cost*, the cost is updated based on the current port costs of the channeling ports. If you change the channel cost, member ports in the channel might be modified and saved to NVRAM. If this is the case, a message appears to list the ports whose port path costs were updated due to the channel cost modification.

## Examples

This example shows how to set the channel 768 path cost to 23:

```
Console> (enable) set channel cost 768 23
Port(s) 1/1-2,7/3,7/5 port path cost are updated to 60.
Channel 768 cost is set to 23.
Warning:channel cost may not be applicable if channel is broken.
Console> (enable)
```

This example shows how to set all channel path costs to 15:

```
Console> (enable) set channel cost all 15
Port(s) 4/1-4 port path cost are updated to 39.
Channel 768 cost is set to 15.
Warning:channel cost may not be applicable if channel is broken.
```

## Related Commands

**show channel**

# set channel vlancost

Use the **set channel vlancost** command to set the channel VLAN cost and automatically adjust the port VLAN costs of the ports in the channel.

**set channel vlancost** *channel\_id cost*

## Syntax Description

<i>channel_id</i>	Number of the channel identification; valid values are from 769 to 896.
<i>cost</i>	Port costs of the ports in the channel.

## Defaults

The default is the VLAN cost is updated automatically based on the current port VLAN costs of the channeling ports.

## Command Types

Switch command.

## Command Modes

Privileged.

## Usage Guidelines

When you do not enter the *cost*, the cost is updated based on the current port VLAN costs of the channeling ports.

You can configure only one channel at a time.

If you change the channel VLAN cost, member ports in the channel might be modified and saved to NVRAM. If this is the case, a message appears to list the ports whose port path costs were updated due to the channel cost modification.

## Examples

This example shows how to set the channel 769 path cost to 10:

```
Console> (enable) set channel vlancost 769 10
Port(s) 1/1-2 vlan cost are updated to 24.
Channel 769 vlancost is set to 10.
Console> (enable)
```

## Related Commands

**show channel**

# set config acl

Use the **set config acl** command to delete the ACL configuration from the NVRAM configuration and save the ACL to a specified file.

```
set config acl {nvram}
```

## Syntax Description

**nvram** Keyword to copy the ACL configuration to NVRAM.

## Defaults

The default is NVRAM.

## Command Types

Switch command.

## Command Modes

Privileged.

## Usage Guidelines

Once the configuration is moved to a Flash file, you must set up the auto-config feature by using the overwrite and append options from the **set boot config-register auto-config** command. You can also set the recurrence on other supervisor engines and switches by using this command.

If you specify multiple configuration files, you must separate the files with a semicolon (;).

If the ACL configuration location is set to **flash**, the following message displays after every commit operation for either Security or QoS:

```
Warning: Use the copy commands to save your ACL configuration to Flash.
```

If you reset the system and there were one or more commits done but no copy commands to one of the files specified in the CONFIG\_FILE variable, the following message displays:

```
Warning: System ACL configuration has been modified but not saved to Flash.
```

## Examples

This example shows how to copy the ACL configuration to the bootflash file:

```
Console> (enable) set config acl flash switchapp.cfg
Upload ACL configuration to bootflash:switchapp.cfg
2843644 bytes available on device bootflash, proceed (y/n) [n]? y
Configuration has been copied successfully.
WARNING: Use the 'set boot config-register auto-config' commands to configure the
auto-config feature.
Console> (enable)
```

This example shows how to copy the ACL configuration to NVRAM:

```
Console> (enable) set config acl nvram
ACL configuration copied to NVRAM.
WARNING: Use the 'set boot config-register auto-config' commands to disable the
auto-config feature.
Console> (enable)
```

■ set config acl

---

**Related Commands**

- set boot config-register
- set boot system flash
- show boot
- copy
- clear config

# set cops

Use the **set cops** command set to configure COPS functionality.

```
set cops server ipaddress [port] [primary] [diff-serv | rsvp]
```

```
set cops domain-name domain_name
```

```
set cops retry-interval initial incr max
```

## Syntax Description

<b>server</b>	Keyword to set the name of the COPS server.
<i>ipaddress</i>	IP address or IP alias of the server.
<i>port</i>	(Optional) Number of the TCP port the switch connects to on the server.
<b>primary</b>	(Optional) Keyword to specify the primary server.
<b>diff-serv</b>	(Optional) Keyword to set the COPS server for differentiated services.
<b>rsvp</b>	(Optional) Keyword to set the COPS server for RSVP+.
<b>domain-name</b> <i>domain_name</i>	Keyword and variable to specify the domain name of the switch.
<b>retry-interval</b>	Keyword to specify the retry interval in seconds.
<i>initial</i>	Initial timeout value; valid values are from 0 to 65535 seconds.
<i>incr</i>	Incremental value; valid values are from 0 to 65535 seconds.
<i>max</i>	Maximum timeout value; valid values are from 0 to 65535 seconds.

## Defaults

The defaults are as follows:

- The retry interval default values are initial = 30 seconds, incr = 30 seconds, max = 5 minutes.
- The default domain-name is a string of length zero.
- No PDP servers are configured.

## Command Types

Switch command.

## Command Modes

Privileged.

## Usage Guidelines

You can configure the names or addresses of up to two PDP servers. One must be the primary, and the optional second server is a secondary, or backup, PDP server.

The COPS domain name can be set globally only; there is no option to set it for each COPS client.

Names such as the server, domain-name, and roles can contain a maximum of 31 characters; longer names are truncated to 31 characters. Valid letters are a-z, A-Z, 0-9, ., - and \_. Names cannot start with an underscore (\_). The names are not case sensitive for matching, but are case sensitive for display.

When specifying the **retry-interval**, the total of the initial timeout value and the incremental value (increment on each subsequent failure) may not exceed the maximum timeout value.

---

**Examples**

This example shows how to configure a server as a primary server:

```
Console> (enable) set cops server 171.21.34.56 primary
171.21.34.56 added to COPS server table as primary server.
Console> (enable)
```

This example shows how to configure a server as a primary RSVP+ server:

```
Console> (enable) set cops server 171.21.34.56 primary rsvp
171.21.34.56 added to COPS server table as primary server for RSVP.
Console> (enable)
```

This example shows how to configure a server as a secondary (or backup) server:

```
Console> (enable) set cops server my_server2
my_server2 added to the COPS server table as backup server.
Console> (enable)
```

This example shows how to set the domain name:

```
Console> (enable) set cops domain-name my_domain
Domain name set to my_domain.
Console> (enable)
```

This example shows how to set the retry interval:

```
Console> (enable) set cops retry-interval 15 1 30
Connection retry intervals set.
Console> (enable)
```

This example shows the display output if the total of the initial timeout value and the incremental value you entered exceeds the maximum timeout value:

```
Console> (enable) set cops retry-interval 15 1 10
The initial timeout plus the increment value may not exceed the max value.
Console> (enable)
```

---

**Related Commands**

**clear cops**  
**show cops**