

restore counters

To restore MAC and port counters, use the **restore counters** command.

restore counters [**all** | *mod/ports*]

Syntax Description	all (Optional) Specifies all ports.
	<i>mod/ports</i> (Optional) Number of the module and the ports on the module.

Defaults This command has no default settings.

Command Types Switch command.

Command Modes Privileged.

Usage Guidelines If you do not specify a range of ports to be restored, then all ports on the switch are restored.

Examples This example shows how to restore MAC and port counters:

```
Console> (enable) restore counters all
This command will restore all counter values reported by the CLI to the hardware counter
values.
Do you want to continue (y/n) [n]? y
MAC and Port counters restored.
Console> (enable)
```

Related Commands [clear counters](#)
[show port counters](#)

rollback

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

```
rollback qos acl {acl_name | all}
```

```
rollback security acl {acl_name | all | adjacency}
```

Syntax Description

qos acl	Specifies QoS ACEs.
<i>acl_name</i>	Name that identifies the VACL whose ACEs are to be affected.
all	Rolls back all ACLs.
security acl	Specifies security ACEs.
adjacency	Rolls back all adjacency tables.

Defaults

This command has no default settings.

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

[commit](#)
[show qos acl info](#)

session

To open a session with a module (for example, the MSM, NAM, or ATM), use the **session** command. This command allows you to use the module-specific CLI.

session *mod*

Syntax Description

<i>mod</i>	Number of the module.
------------	-----------------------

Defaults

This command has no default settings.

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*.

For information on NAM commands, refer to the *Catalyst 6000 Network Analysis Module Installation and Configuration Note*.

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

quit
switch console

set

To display all of the ROM monitor variable names with their values, use the **set** command.

set

Syntax Description This command has no arguments or keywords.

Defaults This command has no default settings.

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

To enable command event accounting on the switch, use the **set accounting commands** command.

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

```
set accounting commands disable
```

Syntax Description

enable	Enables the specified accounting method for commands.
config	Permits accounting for configuration commands only.
enable	Permits accounting for enable mode commands only.
all	Permits accounting for all commands.
stop-only	(Optional) Applies the accounting method at the command end.
tacacs+	Specifies TACACS+ accounting for commands.
disable	Disables 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

To enable accounting of outbound connection events on the switch, use the **set accounting connect** command.

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

```
set accounting connect disable
```

Syntax Description

enable	Enables the specified accounting method for connection events.
start-stop	Applies the accounting method at the start and stop of the connection event.
stop-only	Applies the accounting method at the end of the connection event.
tacacs+	Specifies TACACS+ accounting for connection events.
radius	Specifies RADIUS accounting for connection events.
disable	Disables 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

To enable accounting of normal login sessions on the switch, use the **set accounting exec** command.

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

```
set accounting exec disable
```

Syntax Description	enable	Enables the specified accounting method for normal login sessions.
	start-stop	Specifies the accounting method applies at the start and stop of the normal login sessions.
	stop-only	Specifies the accounting method applies at the end of the normal login sessions.
	tacacs+	Specifies TACACS+ accounting for normal login sessions.
	radius	Specifies RADIUS accounting for normal login sessions.
	disable	Disables 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

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

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

Syntax Description	Parameter	Description
	null-username	Specifies users must have a user ID.
	enable	Enables suppression for a specified user.
	disable	Disables 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

To enable accounting of system events on the switch, use the **set accounting system** command.

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

```
set accounting system disable
```

Syntax Description		
enable	Enables the specified accounting method for system events.	
start-stop	Specifies the accounting method applies at the start and stop of the system event.	
stop-only	Specifies the accounting method applies at the end of the system event.	
tacacs+	Specifies TACACS+ accounting for system events.	
radius	Specifies RADIUS accounting for system events.	
disable	Disables 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)
```

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

To configure the frequency of accounting updates, use the **set accounting update** command.

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

Syntax Description	
new-info	Specifies an update when new information is available.
periodic	Specifies an 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 acllog ratelimit

To limit the number of packets sent to the route processor CPU for bridged ACEs, use the **set acllog ratelimit** command.

set acllog ratelimit *rate*

Syntax Description	<i>rate</i> Number of packets per second; valid values are 1 to 1000. See the “Usage Guidelines” section for more information.
---------------------------	--

Defaults	ACL log rate limiting is disabled.
-----------------	------------------------------------

Command Types	Switch command.
----------------------	-----------------

Command Modes	Privileged.
----------------------	-------------

Usage Guidelines	After entering the set acllog ratelimit command or the clear acllog command, you must either reset the route processor or perform a shut/not shut on the route processor interfaces that have ACEs with the log keyword applied.
-------------------------	---

After entering the **set acllog ratelimit** command, the reset or shut/no shut action causes the bridged ACEs to be redirected to the route processor with rate limiting.

To disable ACL log rate limiting, enter the **clear acllog** command. After entering the **clear acllog** command, the reset or shut/no shut action causes the system to return to its previous behavior. The bridge action remains unchanged.

If the number of packets per second is greater than the rate that you specify, the packets that exceed the specified rate are dropped.

A *rate* value of 500 is recommended.

Examples	This example shows how to enable ACL logging and to specify a rate of 500 for rate limiting:
-----------------	--

```
Console> (enable) set acllog ratelimit 500
```

If the ACLs-LOG were already applied, the rate limit mechanism will be effective on system restart, or after shut/no shut the interface.

```
Console> (enable)
```

Related Commands	clear acllog show acllog
-------------------------	---

set aclmerge algo

To select the ACL merge algorithm, use the **set aclmerge algo** command.

```
set aclmerge algo {bdd | odm}
```

Syntax Description

bdd	Specifies the ACL merge function based on binary decision diagram (BDD).
odm	Specifies the ACL merge function based on order dependent merge (ODM).

Defaults

The merge algorithm is ODM.

Command Types

Switch command.

Command Modes

Privileged.

Usage Guidelines

If BDD is disabled, the merge algorithm can only be ODM. When BDD is enabled, you can choose either the BDD algorithm or the ODM algorithm. Use the **set aclmerge bdd** command to enable or disable BDD.

The ACL merge algorithm that you select is in effect for all new ACL merges. The ACLs already configured are not modified, and they use the ACL merge algorithm that was enabled when the ACLs were merged.

Examples

This example shows how to select ODM as the ACL merge algorithm:

```
Console> (enable) set aclmerge algo odm
Acl merge algorithm set to odm.
Console> (enable)
```

Related Commands

[set aclmerge bdd](#)
[show aclmerge](#)

set aclmerge bdd

To enable or disable the binary decision diagram (BDD) ACL merge algorithm, use the **set aclmerge bdd** command.

```
set aclmerge bdd {enable | disable}
```

Syntax Description

enable	Enables the BDD-based ACL merge function.
disable	Disables the BDD-based ACL merge function.

Defaults

BDD is disabled.

Command Types

Switch command.

Command Modes

Privileged.

Usage Guidelines

When you enable or disable BDD, the change takes effect when your system is restarted.

BDD must be enabled in order to change the ACL merge algorithm.

Enabling BDD on a supervisor engine with 64 MB of RAM could cause memory to run low. To avoid this situation, upgrade the memory or disable BDD.

Examples

This example shows how to disable BDD:

```
Console> (enable) set aclmerge bdd disable
Bdd will be disabled on system restart.
Console> (enable)
```

This example shows how to enable BDD:

```
Console> (enable) set aclmerge bdd enable
Warning:enabling bdd on a supervisor with 64MB RAM
could cause memory to run low, to avoid this situation
please upgrade the memory or disable BDD.
```

```
Bdd will be enabled on system restart.
Console> (enable)
```

Related Commands

[set aclmerge algo](#)
[show aclmerge](#)

set alias

To define aliases (shorthand versions) of commands, use the **set alias** command.

```
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 the *parameter* value, 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

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

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

```
set arp agingtime agingtime
```

Syntax Description	
dynamic	(Optional) Specifies that entries are subject to ARP aging updates.
permanent	(Optional) Specifies that permanent entries are stored in NVRAM until they are removed by the clear arp or clear config command.
static	(Optional) Specifies 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.
agingtime	Sets 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* value, 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

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

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

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

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

```
set authentication enable attempt count [console | telnet]
```

```
set authentication enable lockout time [console | telnet]
```

Syntax	Description
radius	Specifies RADIUS authentication for login.
tacacs	Specifies TACACS+ authentication for login.
kerberos	Specifies Kerberos authentication for login.
enable	Enables the specified authentication method for login.
console	(Optional) Specifies the authentication method for console sessions.
telnet	(Optional) Specifies the authentication method for Telnet sessions.
http	(Optional) Specifies the specified authentication method for HTTP sessions.
all	(Optional) Applies the authentication method to all session types.
primary	(Optional) Specifies the specified authentication method be tried first.
disable	Disables the specified authentication method for login.
local	Specifies local authentication for login.
attempt count	Specifies the number of connection attempts before initiating an error; valid values are 0, from 3 to 10, and 0 to disable.
lockout time	Specifies the lockout timeout; valid values are from 30 to 600 seconds, and 0 to disable.

Defaults Local authentication is enabled for console and Telnet sessions. RADIUS, TACACS+, and Kerberos are disabled for all session types. If authentication is enabled, the default **attempt count** is 3.

Command Types Switch command.

Command Modes Privileged.

Usage Guidelines

Use authentication configuration for both console and Telnet connection attempts unless you use the **console** or **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 as
primary authentication method.
Console> (enable)
```

This example shows how to limit enable mode login attempts:

```
Console> (enable) set authentication enable attempt 5
Enable mode authentication attempts for console and telnet logins set to 5.
Console> (enable)
```

This example shows how to set the enable mode lockout time for both console and Telnet connections:

```
Console> (enable) set authentication enable lockout 50
Enable mode lockout time for console and telnet logins set to 50.
Console> (enable)
```

Related Commands

[set authentication login](#)
[show authentication](#)

set authentication login

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

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

```
set authentication login {radius | tacacs | kerberos} disable [console | telnet | http | all]
```

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

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

```
set authentication login attempt count [console | telnet]
```

```
set authentication login lockout time [console | telnet]
```

Syntax Description	
radius	Specifies the use of the RADIUS server password to determine if you have access permission to the switch.
tacacs	Specifies the use of the TACACS+ server password to determine if you have access permission to the switch.
kerberos	Specifies the Kerberos server password to determine if you have access permission to the switch.
enable	Enables the specified authentication method for login.
console	(Optional) Specifies the authentication method for console sessions.
telnet	(Optional) Specifies the authentication method for Telnet sessions.
http	(Optional) Specifies the authentication method for HTTP sessions.
all	(Optional) Specifies the authentication method for all session types.
primary	(Optional) Specifies that the method specified is the primary authentication method for login.
disable	Disables the specified authentication method for login.
local	Specifies a local password to determine if you have access permission to the switch.
attempt count	Specifies the number of login attempts before initiating an error; valid values are 0, from 3 to 10, and 0 to disable.
lockout time	Specifies the lockout timeout; valid values are from 30 to 43200 seconds, and 0 to disable.

Defaults 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 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)
```

This example shows how to limit login attempts:

```
Console> (enable) set authentication login attempt 5
Login authentication attempts for console and telnet logins set to 5.
Console> (enable)
```

This example shows how to set the lockout time for both console and Telnet connections:

```
Console> (enable) set authentication login lockout 50
Login lockout time for console and telnet logins set to 50.
Console> (enable)
```

Related Commands

[set authentication enable](#)
[show authentication](#)

set authorization commands

To enable authorization of command events on the switch, use the **set authorization commands** command.

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

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

Syntax Description		
enable	Enables the specified authorization method for commands.	
config	Permits authorization for configuration commands only.	
enable	Permits authorization for enable mode commands only.	
all	Permits authorization for all commands.	
<i>option</i>	Switch response to an authorization request; valid values are tacacs+ , if-authenticated , and none . 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 tacacs+ , deny , if-authenticated , and none . See the “Usage Guidelines” section for valid value definitions.	
disable	Disables authorization of command events.	
console	(Optional) Specifies the authorization method for console sessions.	
telnet	(Optional) Specifies the authorization method for Telnet sessions.	
both	(Optional) Specifies the authorization method for both console and Telnet sessions.	

Defaults The default is authorization is disabled.

Command Types Switch command.

Command Modes Privileged.

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

- **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

To enable authorization of privileged mode sessions on the switch, use the **set authorization enable** command.

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

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

Syntax Description	enable	Enables the specified authorization method.
	<i>option</i>	Switch response to an authorization request; valid values are tacacs+ , if-authenticated , and none . 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 tacacs+ , deny , if-authenticated , and none . See the “Usage Guidelines” section for valid value definitions.
	disable	Disables the authorization method.
	console	(Optional) Specifies the authorization method for console sessions.
	telnet	(Optional) Specifies the authorization method for Telnet sessions.
	both	(Optional) Specifies the authorization method for both console and Telnet sessions.

Defaults The default is authorization is disabled.

Command Types Switch command.

Command Modes Privileged.

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

- **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

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

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

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

Syntax Description		
enable		Enables the specified authorization method.
<i>option</i>		Switch response to an authorization request; valid values are tacacs+ , if-authenticated , and none . 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 tacacs+ , deny , if-authenticated , and none . See the “Usage Guidelines” section for valid value definitions.
disable		Disables authorization method.
console		(Optional) Specifies the authorization method for console sessions.
telnet		(Optional) Specifies the authorization method for Telnet sessions.
both		(Optional) Specifies the authorization method for both console and Telnet sessions.

Defaults The default is authorization is denied.

Command Types Switch command.

Command Modes Privileged.

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

- **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 lcd

To configure the Catalyst 6500 series Switch Fabric Module LCD user banner, use the **set banner lcd** command.

```
set banner lcd 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 settings.

Command Types

Switch command.

Command Modes

Privileged.

Usage Guidelines

The user banner cannot contain more than 800 characters, including tabs. Tabs display as eight characters but use only one character of memory.

After you configure the user banner, it is sent to all Catalyst 6500 series Switch Fabric Modules in the switch.

The Switch Fabric Module front panel has a 2 line by 20 character LCD display. To see the LCD user banner, push the SELECT button on the front panel and scroll to the USER CONFIGURATION option. Push the NEXT button to see the user banner.

To clear the LCD user banner, use the **set banner lcd cc** command.

Examples

This example shows how to set the Catalyst 6500 series Switch Fabric Module LCD user banner:

```
Console> (enable) set banner lcd &HelloWorld!&
LCD banner set
Console> (enable)
```

Related Commands

[set banner motd](#)
[set banner telnet](#)
[show banner](#)

set banner motd

To program an MOTD banner to appear before session login, use the **set banner motd** command.

```
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 settings.

Command Types Switch command.

Command Modes Privileged.

Usage Guidelines The MOTD banner cannot contain 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 banner lcd](#)
- [set banner telnet](#)
- [show banner](#)

set banner telnet

To display or suppress the “Cisco Systems Console” Telnet banner message, use the **set banner telnet** command.

set banner telnet {enable | disable}

Syntax Description	enable	Displays the Telnet banner.
	disable	Suppresses the Telnet banner.

Defaults The “Cisco Systems Console” Telnet banner message is enabled.

Command Types Switch.

Command Modes Privileged.

Examples This example shows how to display the Telnet banner message:

```
Console> (enable) set banner telnet enable
Cisco Systems Console banner will be printed at telnet.
Console> (enable)
```

This example shows how to suppress the Telnet banner message:

```
Console> (enable) set banner telnet disable
Cisco Systems Console banner will not be printed at telnet.
Console> (enable)
```

Related Commands

- [set banner lcd](#)
- [set banner motd](#)
- [show banner](#)

set boot auto-config

To specify one or more configuration files to use to configure the switch at bootup, use the **set boot auto-config** command. 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

To configure the boot configuration register value, use the **set boot config-register** command.

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

```
set boot config-register baud {1200 | 2400 | 4800 | 9600 | 19200 | 38400} [mod]
```

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

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

Syntax	Description
0xvalue	Sets the 16-bit configuration register value.
<i>mod</i>	(Optional) Module number of the supervisor engine containing the Flash device.
baud 1200 2400 4800 9600 19200 38400	Specifies the console baud rate.
ignore-config	Sets the ignore-config feature.
enable	Enables the specified feature.
disable	Disables the specified feature.
boot	Specifies the boot image to use on the next restart.
rommon	Specifies booting from the ROM monitor.
bootflash	Specifies booting from the bootflash.
system	Specifies 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 with 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

[copy](#)
[set boot auto-config](#)
[set boot system flash](#)
[set config acl nvram](#)
[show boot](#)
[show config](#)

set boot config-register auto-config

To configure auto-config file dispensation, use the **set boot config-register auto-config** command.

```
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

recurring	Sets 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.
non-recurring	Sets 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.
overwrite	Causes the auto-config file to overwrite the NVRAM configuration.
append	Causes the auto-config file to append to the file currently in the NVRAM configuration.
sync enable disable	Enables or disables 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 files on the supervisor engine, the files will also be deleted on the standby supervisor engine.

If you enter the **sync enable** keywords, this enables synchronization to force the configuration files to synchronize automatically to the redundant supervisor engine. The files are kept consistent with what is on 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** command 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

To set the Network Analysis Module (NAM) or Intrusion Detection System (IDS) boot environment, use the **set boot device** command.

```
set boot device bootseq[,bootseq] mod [mem-test-full]
```

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.	
mem-test-full	Specifies a full memory test.	

Defaults The default is a partial memory test.

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 entering the *bootseq*, use the format *bootdevice[:bootdevice-qualifier]* where:

- *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 valid values for **pcmcia** are slot0 or slot1.

The colon between *bootdevice* and *bootdevice-qualifier* is required.

You can enter multiple *bootseqs* 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 stores the boot device list in NVRAM.

This command is supported by the NAM or IDS 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 sync now

To immediately initiate synchronization of the system image between the active and redundant supervisor engine, use the **set boot sync now** command.

set boot sync now

Syntax Description This command has no arguments or keywords.

Defaults The default is synchronization is disabled.

Command Types Switch command.

Command Modes Privileged.

Usage Guidelines The **set boot sync now** command is similar to the **set boot config-register auto-config** command with the **sync** keyword added. The **set boot sync now** command initiates synchronization to force the configuration files to synchronize automatically to the redundant supervisor engine. The files are kept consistent with what is on the active supervisor engine.

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

```
Console> (enable) set boot sync now
Console> (enable)
```

Related Commands [set boot auto-config](#)
[show boot](#)

set boot system flash

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

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

Syntax Description	
<i>device</i> :	Device where the Flash resides.
<i>filename</i>	(Optional) Name of the configuration file.
prepend	(Optional) Places 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 settings.

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 problem-free 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)
```

Related Commands

[clear boot system](#)
[show boot](#)

set cam

To add entries into the CAM table, set the aging time for the CAM table, and configure traffic filtering from and to a specific host, use the **set cam** command.

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

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

```
set cam { static | permanent } filter { unicast_mac } vlan
```

```
set cam agingtime vlan agingtime
```

Syntax Description		
dynamic		Specifies entries are subject to aging.
static		Specifies entries are not subject to aging.
permanent		Specifies permanent entries are stored in NVRAM until they are removed by the clear cam or clear config 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; valid values are from 1 to 1005 and from 1025 to 4094.
<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.
filter		Specifies a traffic filter entry.
agingtime		Sets the period of time after which an entry is removed from the table.
<i>agingtime</i>		Number of seconds (0 to 1,000,000) dynamic entries remain in the table before being deleted.

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 supervisor engine). The default aging time for all configured VLANs is 300 seconds.

The *vlan* variable is required when you configure the traffic filter entry.

Setting the aging time to 0 disables aging.

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

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 MAC address and VLAN for a host can be stored in the NVRAM it is maintained even after a reset.

The *vlan* value 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 remain in the table until you reset the active supervisor engine.

Enter the *route_descr* variable as two hexadecimal bytes in the following format: 004F. Do not use a “-” to separate the bytes.

**Note**

Static CAM entries that are configured on the active supervisor engine are lost after fast switchover. You must reconfigure CAM entries after fast switchover.

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)
```

This example shows how to add a traffic filter entry to the table:

```
Console> (enable) set cam static filter 00-02-03-04-05-06 1
Filter entry added to CAM table.
Console> (enable)
```

Related Commands

[clear cam](#)
[show cam](#)

set cam notification

To enable notification when a MAC address change occurs to the CAM table and to set the time between notifications, use the **set cam notification** command.

set cam notification { **enable** | **disable** }

set cam notification { **added** | **removed** } { **enable** | **disable** } { *mod/port* }

set cam notification historysize *log_size*

set cam notification interval *time*

set cam notification move { **enable** | **disable** }

set cam notification threshold { **enable** | **disable** }

set cam notification threshold limit *percentage*

set cam notification threshold interval *time*

Syntax Description

enable	Enables notification that a change has occurred.
disable	Disables notification that a change has occurred.
added	Specifies notification when a MAC address is learned.
removed	Specifies notification when a MAC address is deleted.
<i>mod/port</i>	Number of the module and the port.
historysize	Creates a notification history log.
<i>log_size</i>	Number of entries in the notification history log; valid sizes are between 0 and 500 entries.
interval	Sets the maximum wait time between notifications.
<i>time</i>	Time between notification; valid values are greater than or equal to 0 (specified in seconds).
move	Specifies MAC move notifications.
threshold	Sets parameters for CAM usage monitoring
limit	Sets CAM usage monitoring percentage.
<i>percentage</i>	Percentage of usage monitoring.

Defaults

By default, notification is disabled.

By default, the interval time is set to 1 second.

By default, the history size is set to 1 entry.

Command Types

Switch command.

Command Modes Privileged.

Usage Guidelines You can globally disable notifications using the **set cam notification disable** command, but the other notification configuration settings will remain configured. The notification configuration settings can be reset using the **clear config** command. The **clear cam notification** command can be used to clear the history log or reset notification counters.

If you set the interval time to 0, the switch will send notifications immediately. There is an impact on the performance of the switch when you set the interval time to zero (0).

You can configure the switch to generate MAC notification SNMP traps using the **set snmp enable macnotification** command. MAC notification SNMP traps are generated even when the history log size is set to zero (0).

Examples This example shows how to enable notification when a MAC address change occurs to the CAM table:

```
Console> (enable) set cam notification enable
MAC address change detection globally enabled
Be sure to specify which ports are to detect MAC address changes
with the 'set cam notification [added|removed] enable <m/p>' command.
SNMP traps will be sent if 'set snmp trap enable macnotification' has been set.
Console> (enable)
```

This example shows how to enable notification when a new MAC address is added to ports 1-4 on module 3 in the CAM table:

```
Console> (enable) set cam notification added enable 3/1-4
MAC address change notifications for added addresses are
enabled on port(s) 3/1-4
Console> (enable)
```

This example shows how to enable notification when a new MAC address is added to the CAM table on ports 1-4 on module 2:

```
Console> (enable) set cam notification added enable 2/1-4
MAC address change notifications for added addresses are
enabled on port(s) 2/1-4
Console> (enable)
```

This example shows how to enable notification when a MAC address is deleted from the CAM table of ports 3-6 on module 3:

```
Console> (enable) set cam notification removed enable 3/3-6
MAC address change notifications for removed addresses are
enabled on port(s) 3/3-6
```

This example shows how to set the history log size to 300 entries:

```
Console> (enable) set cam notification historysize 300
MAC address change history log size set to 300 entries
Console> (enable)
```

This example shows how to set the interval time to 10 seconds between notifications:

```
Console> (enable) set cam notification interval 10
MAC address change notification interval set to 10 seconds
Console> (enable)
```

■ set cam notification

Related Commands

[clear cam](#)
[clear cam notification](#)
[set cam](#)
[set snmp trap](#)
[show cam](#)
[show cam notification](#)

set cdp

To enable, disable, or configure Cisco Discovery Protocol (CDP) features globally on all ports or on specified ports, use the **set cdp** command.

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

```
set cdp interval interval
```

```
set cdp holdtime holdtime
```

```
set cdp version v1 | v2
```

```
set cdp format device-id {mac-address | other}
```

Syntax Description		
enable		Enables the CDP feature.
disable		Disables the CDP feature.
<i>mod/ports..</i>		Number of the module and the ports on the module.
interval		Specifies 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.
holdtime		Specifies 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.
version v1 v2		Specifies the CDP version number.
format device-id		Sets the device-ID TLV format.
mac-address		Specifies that the device-ID TLV carry the MAC address of the sending device in ASCII, in canonical format.
other		Specifies that the device's hardware serial number concatenated with the device name between parenthesis.

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 you globally enable CDP, 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/ports...* value 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)
```

This example shows how to set the device ID format to MAC address:

```
Console> (enable) set cdp format device-id mac-address
Device Id format changed to MAC-address
Console> (enable)
```

Related Commands

[show cdp](#)
[show port cdp](#)

set channelprotocol

To set the protocol that manages channeling on a module, use the **set channelprotocol** command.

```
set channelprotocol { pagp | lacp } mod
```

Syntax Description		
	pagp	Specifies PAgP.
	lacp	Specifies LACP.
	<i>mod</i>	Number of the module.

Defaults The default for the channel protocol is PAgP.

Command Types Switch command.

Command Modes Privileged.

Usage Guidelines LACP is supported on all Ethernet interfaces.

PAgP and LACP manage channels differently. When all the ports in a channel get disabled, PAgP removes them from its internal channels list; **show** commands do not display the channel. With LACP, when all the ports in a channel get disabled, LACP does not remove the channel; **show** commands continue to display the channel even though all its ports are down. To determine if a channel is actively sending and receiving traffic with LACP, use the **show port** command to see if the link is up or down.

LACP does not support half-duplex links. If a port is in active/passive mode and becomes half duplex, the port is suspended (and a syslog message is generated). The port is shown as “connected” using the **show port** command and as “not connected” using the **show spantree** command. This discrepancy is because the port is physically connected but never joined spanning tree. To get the port to join spanning tree, either set the duplex to full or set the channel mode to off for that port.

For more information about PAgP and LACP, refer to the “Configuring EtherChannel” chapter of the *Catalyst 6500 Series Switch Software Configuration Guide*.

Examples This example shows how to set PAgP for module 3:

```
Console> (enable) set channelprotocol pagp 3
Channeling protocol set to PAGP for module(s) 3.
Console> (enable)
```

This example shows how to set LACP for modules 2, 4, 5, and 6:

```
Console> (enable) set channelprotocol lacp 2,4-6
Channeling protocol set to LACP for module(s) 2,4,5,6.
Console> (enable)
```

Related Commands

[clear lacp-channel statistics](#)
[set lacp-channel system-priority](#)
[set port lacp-channel](#)
[set spantree channelcost](#)
[set spantree channelvlancost](#)
[show channelprotocol](#)
[show lacp-channel](#)

set channel vlancost

To set the channel VLAN cost, use the **set channel vlancost** command.

```
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.



Note

The **set channel vlancost** command creates a “set spantree portvlancost” entry for each port in the channel. You must then manually reenter the **set spantree portvlancost** command for at least one port in the channel, specifying the VLAN or VLANs that you want associated with the port. When you associate the desired VLAN or VLANs with one port, all ports in the channel are automatically updated. Refer to Chapter 6, “Configuring EtherChannel,” in the *Catalyst 6500 Series Switch Software Configuration Guide* for more information.



Note

With software releases 6.2(1) and earlier, the 6- and 9-slot Catalyst 6500 series switches support a maximum of 128 EtherChannels.

With software releases 6.2(2) and later, due to the port ID handling by the spanning tree feature, the maximum supported number of EtherChannels is 126 for a 6- or 9-slot chassis and 63 for a 13-slot chassis. Note that the 13-slot chassis was first supported in software release 6.2(2).

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)
```

After you enter this command, you must reenter the [set spantree portvlancost](#) command so that the desired VLAN or VLANs are associated with all the channel ports.

This example shows how to associate the channel 769 path cost to 10 for VLAN 1 through VLAN 1005:

```
Console> (enable) set spantree portvlancost 1/1 cost 24 1-1005
Port 1/1 VLANs 1025-4094 have path cost 19.
Port 1/1 VLANs 1-1005 have path cost 24.
Port 1/2 VLANs 1-1005 have path cost 24.
Console> (enable)
```

Related Commands

[set spantree portvlancost](#)
[show channel](#)

set config acl nvram

To copy the current committed ACL configuration from DRAM back into NVRAM, use the **set config acl nvram** command.

set config acl nvram

Syntax Description This command has no arguments or keywords.

Defaults The default is NVRAM.

Command Types Switch command.

Command Modes Privileged.

Usage Guidelines This command fails if there is not enough space in NVRAM.

This command copies the current committed configuration to NVRAM; this configuration might be different from the configuration in the auto-config file. After the ACL configuration is copied into NVRAM, you must turn off the auto-config options using the **clear boot auto-config** command.

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

```
Console> (enable) set config acl nvram  
ACL configuration copied to NVRAM.  
Console> (enable)
```

Related Commands

- [clear config](#)
- [copy](#)
- [set boot config-register](#)
- [set boot system flash](#)
- [show boot](#)

set config mode

To change the configuration mode from a binary model to a text model or to automatically save the system configuration in text mode in NVRAM, use the **set config mode** command.

set config mode binary

set config mode text { **nvr**am | *device:file-id* }

set config mode text auto-save { **enable** | **disable** }

set config mode text auto-save interval [*mins*]

Syntax Description		
binary	Sets the system configuration mode to a binary model.	
text	Sets the system configuration mode to a text model.	
nvr am	Specifies the saved configuration be stored in NVRAM.	
<i>device:file-id</i>	Name of the device and filename where the saved configuration will be stored.	
auto-save	Specifies saving the text configuration in NVRAM automatically.	
enable	Enables saving the text configuration in NVRAM automatically.	
disable	Disables saving the text configuration in NVRAM automatically.	
interval	Sets the time interval between occurrences of saving the text configuration in NVRAM; see the “Usage Guidelines” section for more information.	
<i>mins</i>	(Optional) Number of minutes between occurrences of saving the text configuration in NVRAM; valid values are from 30 minutes to 35000 minutes (25 days).	

Defaults

The default setting of this command is binary, saving the configuration to NVRAM.

The number of minutes between occurrences of saving the text configuration in NVRAM is 30 minutes.

Command Types

Switch command.

Command Modes

Privileged.

Usage Guidelines

You can specify the time interval between occurrences of saving the text configuration in NVRAM even if the system is in binary mode. If you do not specify the number of minutes after entering the **interval** keyword, the interval is set to the default of 30 minutes.

The text configuration is not saved automatically in NVRAM unless the auto-save feature is enabled. To enable the auto-save feature, you must first set the system configuration mode to text and configure the system to save the text configuration in NVRAM. If the system configuration mode is set to a binary model, you cannot enable the auto-save feature.

In release 7.6(10) and later releases, the maximum value that you can set for the *mins* value is 35000 minutes (25 days). Any commands with an interval configuration that is greater than 35000 minutes in the configuration file when loaded on the switch is set to the maximum value of 35000 minutes internally.

Examples

This example shows how to set the configuration mode to binary:

```
Console> (enable) set config mode binary
System configuration copied to NVRAM. Configuration mode set to binary.
Console> (enable)
```

This example shows how to set the configuration mode to text and designate the location and filename for saving the text configuration file:

```
Console> (enable) set config mode text bootflash:switch.cfg
Binary system configuration has been deleted from NVRAM. Configuration mode set to text.
Use the write memory command to save configuration changes. System configuration file set
to: bootflash:switch.cfg
The file specified will be used for configuration during the next bootup.
Console> (enable)
```

This example shows how to enable the auto-save feature when the configuration is set to text mode and the system is configured to save the text configuration in NVRAM:

```
Console> (enable) set config mode text auto-save enable
auto-save feature has been enabled
auto-save feature has started
Please do a write mem manually if you plan to reboot the switch or any card before first
expiry of the timer
Console> (enable)
```

This example shows the message that is displayed if you attempt to enable the auto-save feature when the configuration is not set to text mode and the system is not configured to save the text configuration in NVRAM:

```
Console> (enable) set config mode text auto-save enable
auto-save cannot be enabled unless config mode is set to text and config file is stored in
nvram.
Use the 'set config mode text nvram' command to enable automatic saving of the system
configuration to nvram
Console> (enable)
```

This example shows how to set the interval between saves to 2880 minutes:

```
Console> (enable) set config mode text auto-save interval 2880
auto-save interval set to 2880 minutes
Console> (enable)
```

This example shows how to set the interval between saves to the default setting of 30 minutes:

```
Console> (enable) set config mode text auto-save interval
auto-save interval set to 30 minutes
Console> (enable)
```

Related Commands

[show config mode](#)
[write](#)

set cops

To configure COPS functionality, use the **set cops** command.

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

set cops domain-name *domain_name*

set cops retry-interval *initial incr max*

Syntax Description

server	Sets 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.
primary	(Optional) Specifies the primary server.
diff-serv	(Optional) Sets the COPS server for differentiated services.
rsvp	(Optional) Sets the COPS server for RSVP+.
domain-name <i>domain_name</i>	Specifies the domain name of the switch.
retry-interval	Specifies 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 policy decision point (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](#)

set crypto key rsa

To generate and configure an RSA key pair, use the **set crypto key rsa** command.

```
set crypto key rsa nbits [force]
```

Syntax Description	<i>nbits</i>	Size of the key; valid values are 512 to 2048 bits.
	force	(Optional) Regenerates the keys and suppress the warning prompt of overwriting existing keys.

Defaults The command has no default settings.

Command Types Switch command.

Command Modes Privileged.

Usage Guidelines The **crypto** commands are supported on systems that run these image types only:

- supk9 image—for example, cat6000-supk9.6-1-3.bin
- supcvk9 image—for example, cat6000-supcvk9.6-1-3.bin

If you do not enter the **force** keyword, the **set crypto key** command is saved into the configuration file and you will have to use the **clear config all** command to clear the RSA keys.

The *nbits* value is required.

To support SSH login, you first must generate an RSA key pair.

Examples This example shows how to create an RSA key:

```
Console> (enable) set crypto key rsa 1024
Generating RSA keys.... [OK]
Console> (enable)
```

Related Commands [clear crypto key rsa](#)
[show crypto key](#)