CHAPTER 2

Cisco IOS Commands for the Catalyst 6500 Series Switches with the Supervisor Engine 32 PISA

This chapter contains an alphabetical listing of Cisco IOS commands that are unique to the Catalyst 6500 series switches that are configured with the Supervisor Engine 32 and the Programmable Intelligent Services Accelerator (PISA). For information about Cisco IOS commands that are not contained in this publication, refer to the current Cisco IOS documentation including:

- *Cisco IOS Release 12.2 Configuration Fundamentals Configuration Guide*
- *Cisco IOS Release 12.2 Command Reference*

action

To set the packet action clause, use the **action** command. To remove an action clause, use the **no** form of this command.

action {{**drop** [**log**]} | {**forward** [**capture**]} | {**redirect** {*interface interface-number*}} | {**port-channel** *channel-id*} {*interface interface-number*} | {**port-channel** *channel-id*} **...**}

no action {{**drop** [**log**]} | {**forward** [**capture**]} | {**redirect** {*interface interface-number*}} | {**port-channel** *channel-id*} {*interface interface-number*} | {**port-channel** *channel-id*} **...**}

Syntax Description

drop	Drops the packets.
log	(Optional) Logs the dropped packets in the software.
forward	Forwards (switched by hardware) the packets to its destination.
capture	(Optional) Sets the capture bit for the forwarded packets so that ports with the capture function enabled also receive the packets.
redirect *interface*	Redirects packets to the specified interfaces; possible valid values are **fastethernet, gigabitethernet, and tengigabitethernet**. See the "Usage Guidelines" section for additional valid values.
interface-number	Module and port number; see the "Usage Guidelines" section for valid values.
port-channel *channel-id*	Specifies the port channel to redirect traffic; valid values are a maximum of 64 values ranging from 1 to 256.

Defaults

This command has no default settings.

Command Modes

VLAN access-map submode

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

The *interface-number* argument designates the module and port number. Valid values for *interface-number* depend on the specified interface type and the chassis and module that are used. For example, if you specify a Gigabit Ethernet interface and have a 48-port 10/100BASE-T Ethernet module that is installed in a 13-slot chassis, valid values for the module number are from 1 to 13 and valid values for the port number are from 1 to 48.

Each redirect action allows you to specify a list of up to five destination interfaces. There is also a limit of up to 255 different interface lists that can be used by redirect actions.

The redirect action supports interface lists instead of single interfaces as shown in the following example:

[. . .] {**redirect** {{**ethernet** | **gigabitethernet** | **tengigabitethernet**} *slot*/port} | {**port-channel** *channel-id*}}

The action clause specifies the action to be taken when a match occurs.

The forwarded packets are subject to any applied Cisco IOS ACLs. The **capture** keyword sets the capture bit in VACL-forwarded packets. Ports with the capture function enabled can receive VACL-forwarded packets that have the capture bit set. Only VACL-forwarded packets that have the capture bit set can be captured.

When the **log** keyword is specified, dropped packets are logged in the software. Only dropped IP packets can be logged. The **redirect** keyword allows you to specify up to five interfaces, which can be physical interfaces or EtherChannels. An EtherChannel member is not allowed to be a redirect interface.

VACLs on WAN interfaces support only the **action forward capture** command.

The action clause in a VACL can be forward, drop, capture, or redirect. Traffic can also be logged. VACLs applied to WAN interfaces do not support the redirect or log actions.

The redirect interface must be in the VLAN for which the VACL map is configured.

In a VLAN access map, if at least one ACL is configured for a packet type (IP, IPX, or MAC), the default action for the packet type is **drop** (deny).

If an ACL is not configured for a packet type, the default action for the packet type is **forward** (permit).

If an ACL for a packet type is configured and the ACL is empty or undefined, the configured action will be applied to the packet type.

Examples

This example shows how to define a drop and log action:

```
Router(config-access-map)# action drop log
Router(config-access-map)#
```

This example shows how to define a forward action:

```
Router(config-access-map)# action forward
Router(config-access-map)#
```

Related Commands

Command	Description
match	Specifies the match clause by selecting one or more ACLs for a VLAN access-map sequence.
show vlan access-map	Displays the contents of a VLAN-access map.
vlan access-map	Creates a VLAN access map or enters the VLAN access-map command mode.

apply

To implement the proposed new VLAN database, increment the database configuration number, save it in NVRAM, and propagate it throughout the administrative domain, use the **apply** command.

apply

Syntax Description

This command has no arguments or keywords.

Defaults

This command has no default settings.

Command Modes

VLAN configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

The **apply** command implements the configuration changes that you made after you entered VLAN database mode and uses them for the running configuration. This command keeps you in VLAN database mode.

You cannot use this command when the Catalyst 6500 series switch is in the VTP client mode.

You can verify that VLAN database changes have occurred by entering the **show vlan** command in privileged EXEC mode.

Examples

This example shows how to implement the proposed new VLAN database and recognize it as the current database:

```
Router(config-if-vlan)# apply
Router(config-if-vlan)#
```

Related Commands

Command	Description
abort	Abandons the proposed new VLAN database.
exit	Implements the proposed new VLAN database.
reset	Leaves the proposed new VLAN database, remains in VLAN configuration mode, and resets the new database so that it is identical to the current VLAN database.
show vlan	Displays VLAN information.
shutdown vlan	Shuts down local traffic on a specified VLAN.
vtp	Configures the global VTP state.

arp access-list

To configure an ARP ACL for ARP inspection and QoS filtering and enter the ARP ACL configuration submode, use the **arp access-list** command. To remove the ARP ACL, use the **no** form of this command.

arp access-list *name*

no arp access-list *name*

Syntax Description

name	Name of the access list.

Defaults

This command has no default settings.

Command Modes

Global configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

Once you are in the ARP ACL configuration submode, you can add **permit** or **deny** clauses to permit or deny QoS to the flows. The following syntax is available in the ARP QoS ACL configuration submode for QoS filtering; all other configurations will be rejected at the time of the policy-map attachment to the interfaces:

{**permit** | **deny**} {**ip** {**any** | {**host** *sender-ip* [*sender-ip-mask*]}}} {**mac any**}

no {**permit** | **deny**} {**ip** {**any** | {**host** *sender-ip* [*sender-ip-mask*]}}} {**mac any**}

permit	Specifies to apply QoS to the flows.
deny	Skips the QoS action that is configured for traffic matching this ACE.
ip	Specifies the IP ARP packets.
any	Specifies any IP ARP packets.
host *sender-ip*	Specifies the IP address of the host sender.
sender-ip-mask	(Optional) Wildcard mask of the host sender.
mac any	Specifies MAC-layer ARP traffic.
no	Deletes an ACE from an ARP ACL.

Once you are in the ARP ACL configuration submode, the following configuration commands are available for ARP inspection:

- **default**—Sets a command to its defaults. You can use the **deny** and **permit** keywords and arguments to configure the default settings.
- **deny**—Specifies the packets to reject.

- **exit**—Exits the ACL configuration mode.
- **no**—Negates a command or sets its defaults.
- **permit**— Specifies the packets to forward.

You can enter the **permit** or **deny** keywords to configure the permit or deny clauses to forward or drop ARP packets based on some matching criteria. The syntax for the **permit** and **deny** keywords are as follows:

{**permit** | **deny**} **ip** {**any** | {**host** {*sender-ip* | {*sender-ip sender-ip-mask*}}}} **mac** {**any** | {**host** {*sender-mac* | {*sender-mac sender-mac-mask*}}}} [**log**]

{**permit** | **deny**} **request ip** {**any** | {**host** {*sender-ip* | {*sender-ip sender-ip-mask*}}}} **mac** {**any** | {**host** {*sender-mac* | {*sender-mac sender-mac-mask*}}}} [**log**]

{**permit** | **deny**} **response ip** {**any** | {**host** {*sender-ip* | {*sender-ip sender-ip-mask*}}}} [{**any** | {**host** {*target-ip* | {*target-ip target-ip-mask*}}}}] **mac** {**any** | {**host** {*sender-mac* | {*sender-mac sender-mac-mask*}}}} [**any** | {**host** {*target-mac* | {*target-mac target-mac-mask*}}]} [**log**]

permit	Specifies packets to forward.
deny	Specifies packets to reject.
ip	Specifies the sender IP address.
any	Specifies any sender IP address.
host	Specifies a single sender host.
sender-ip	IP address of the host sender.
sender-ip-mask	Wildcard mask of the host sender.
mac any	Specifies any MAC address.
mac host	Specifies a single sender host MAC address.
sender-mac	MAC address of the host sender.
sender-mac-mask	Wildcard mask of the host sender.
log	(Optional) Specifies log on match.
request	Specifies ARP requests.
response	Specifies ARP responses.
any	(Optional) Specifies any target address.
host	(Optional) Specifies a single target host.
target-mac	MAC address of the target host.
target-mac-mask	Subnet mask of the target host.

If you enter the **ip** keyword without the **request** or **response** keywords, the configuration applies to both requests and responses.

Once you define an ARP ACL, you can apply it to VLANs using the **ip arp inspection filter** command for ARP inspection.

Incoming ARP packets are compared against the ARP access list, and packets are permitted only if the access list permits them. If access lists deny packets because of explicit denies, they are dropped. If packets get denied because of the implicit deny, they are matched against the list of DHCP bindings, unless the access list is static or the packets are not compared against the bindings.

When an ARP access list is applied to a VLAN for dynamic ARP inspection, the ARP packets containing only IP-to-Ethernet MAC bindings are compared against the ACLs. All other packet types are bridged in the incoming VLAN without any validation.

ACL entries are scanned in the order that you enter them. The first matching entry is used. To improve performance, place the most commonly used entries near the beginning of the ACL.

An implicit **deny any ip mac any** entry exists at the end of an ACL unless you include an explicit **permit ip any mac any** entry at the end of the list.

All new entries to an existing list are placed at the end of the list. You cannot add entries to the middle of a list.

Examples

This example shows how to create a new ARP ACL or enter the submode of an existing ARP ACL:

```
Router(config)# arp access-list arpacl22
Router(config-arp-nacl)#
```

This example shows how to create an ARP ACL named arp_filtering that denies QoS but permits MAC-layer ARP traffic:

```
Router(config)# arp access-list arp_filtering
Router(config-arp-nacl)# permit ip host 1.1.1.1 mac any
Router(config-arp-nacl)# deny any ip mac any
Router(config-arp-nacl)#
```

Related Commands

Command	Description
show arp	Displays information about the ARP table.

attach

To connect to a specific module from a remote location, use the **attach** command.

attach *num*

Syntax Description

num	Module number; see the "Usage Guidelines" section for valid values.

Defaults

This command has no default settings.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

Caution When you enter the **attach** or **remote login** command to access another console from your switch, if you enter global or interface configuration mode commands, the switch might reset.

The valid values for *num* depend on the chassis that is used. For example, if you have a 13-slot chassis, valid values for the module number are from 1 to 13.

This command is supported on the supervisor engine only.

When you execute the **attach** *num* command, the prompt changes to Switch-sp#.

The **attach** command is identical to the **remote login module** *num* command.

There are two ways to end this session:

- You can enter the **exit** command as follows:

```
Switch-sp# exit

[Connection to Switch closed by foreign host]
Router#
```

- You can press **Ctrl-C** three times as follows:

```
Switch-sp# ^C
Switch-sp# ^C
Switch-sp# ^C
Terminate remote login session? [confirm] y
[Connection to Switch closed by local host]
Router#
```

Examples

This example shows how to log in remotely to the supervisor engine:

```
Router# attach 5
Trying Switch ...
Entering CONSOLE for Switch
Type "^C^C^C" to end this session
Terminate remote login session? [confirm] yes
[Connection to Switch closed by local host]

Switch-sp#
```

Related Commands

Command	Description
remote login	Accesses the Catalyst 6500 series switch console or a specific module.

auto-sync

To enable automatic synchronization of the configuration files in NVRAM, use the **auto-sync** command. To disable automatic synchronization, use the **no** form of this command.

auto-sync {**startup-config** | **config-register** | **bootvar** | **running-config** | **standard**}

no auto-sync {**startup-config** | **config-register** | **bootvar** | **standard**}

Syntax Description

startup-config	Specifies the automatic synchronization of the startup configuration.
config-register	Specifies the automatic synchronization of the configuration register configuration.
bootvar	Specifies the automatic synchronization of the BOOTVAR configuration.
running-config	Specifies the automatic synchronization of the running configuration.
standard	Specifies the automatic synchronization of the startup-config, BOOTVAR, and configuration registers.

Defaults

Automatic synchronization of the running configuration**.**

Command Modes

Main-cpu redundancy

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

If you enter the **no auto-sync standard** command, no automatic synchronizations occur. If you want to enable any of the keywords, you have to enter the appropriate command for each keyword.

Examples

This example shows how (from the default configuration) to enable automatic synchronization of the configuration register in the main CPU:

```
Router# configure terminal
Router (config)# redundancy
Router (config-r)# main-cpu
Router (config-r-mc)# no auto-sync standard
Router (config-r-mc)# auto-sync config-register
Router (config-r-mc)#
```

Related Commands

Command	Description
redundancy	Enters redundancy configuration mode.

bgp regexp deterministic

To configure Cisco IOS software to use the deterministic processing time regular expression engine, use the **bgp regexp deterministic** command. To configure Cisco IOS software to use the default regular expression engine, use the **no** form of this command.

bgp regexp deterministic

no bgp regexp deterministic

Syntax Description

This command has no keywords or arguments.

Defaults

The default regular expression engine is enabled.

Command Modes

Router configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

The default Cisco IOS regular expression engine uses a recursive algorithm. This engine is effective but uses more system resources as the complexity of regular expressions increases. The recursive algorithm works well for simple regular expressions, but is less efficient when processing very complex regular expressions because of the backtracking that is required by the default engine to process partial matches. In some cases, CPU watchdog timeouts and stack overflow traces have occurred because of the length of time that the default engine requires to process very complex regular expressions.

The deterministic processing time regular expression engine does not replace the default regular expression engine. The new engine employs an improved algorithm that eliminates excessive backtracking and greatly improves performance when processing complex regular expressions. When the new engine is enabled, complex regular expressions are evaluated more quickly, and CPU watchdog timeouts and stack overflow traces will not occur. However, the new regular expression engine takes longer to process simple regular expressions than the default engine.

We recommend that you use the new regular expression engine if you need to evaluate complex regular expressions or if you have observed problems related to evaluating regular expressions. We recommend that you use the default regular expression engine if you use only simple regular expressions. The new engine can be enabled by entering the **bgp regexp deterministic** command under a BGP routing process. The default regular expression engine can be reenabled by entering the **no** form of this command.

Examples

This example shows how to configure Cisco IOS software to use the deterministic processing time regular expression engine:

```
Router(config)# router bgp 1
Router(config-router)# bgp regexp deterministic
Router(config-router)#
```

This example shows how to configure Cisco IOS software to use the default regular expression engine:

```
Router(config)# router bgp 1
Router(config-router)# no bgp regexp deterministic
Router(config-router)#
```

boot config

To specify the device and filename of the configuration file from which the system configures itself during initialization (startup), use the **boot config** command. To remove the specification, use the **no** form of this command.

boot config {*device***:***file-name*}

no boot config

Syntax Description

*device***:**	Device identification; see the "Usage Guidelines" section for a list of the valid values.
file-name	Configuration filename.

Defaults

The configuration file is located in NVRAM.

Command Modes

Global configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

The valid values for *device***:** are as follows:

- **disk0:**
  - One external CompactFlash Type II slot
  - Supports CompactFlash Type II flash PC cards
- **sup-bootdisk:**
  - Supervisor Engine 32 256-MB internal CompactFlash flash memory
  - From the Supervisor Engine 32 ROMMON, it is bootdisk:
- **bootdisk:**
  - PISA 256-MB internal CompactFlash flash memory
  - Not accessible from the Supervisor Engine 32 ROMMON

When you use the **boot config** command, you affect only the running configuration. You must save the environment variable setting to your startup configuration to place the information under ROM monitor control and to have the environment variable function as expected. Use the **copy system:running-config nvram:startup-config** command to save the environment variable from your running configuration to your startup configuration.

The software displays an error message and does not update the CONFIG_FILE environment variable in the following situations:

- You specify **nvram:** as the file system, and it contains only a distilled version of the configuration. (A distilled configuration does not contain access lists.)
- You specify a configuration file in the filename argument that does not exist or is not valid.

During initialization, the NVRAM configuration is used when the CONFIG_FILE environment variable does not exist or when it is null (such as at a first-time startup). If the software detects a problem with NVRAM or the configuration it contains, the device enters setup mode.

When you use the **no** form of this command, the NVRAM configuration is used as the startup configuration.

You can view the contents of the BOOT, BOOTLDR, and the CONFIG_FILE environment variables using the **show bootvar** command. This command displays the settings for these variables as they exist in the startup configuration as well as in the running configuration if a running configuration setting differs from a startup configuration setting.

Examples

This example shows how to set the configuration file that is located in the internal flash memory to configure itself during initialization. The third line copies the specification to the startup configuration, ensuring that this specification takes effect upon the next reload.

```
Router (config)# boot config disk0:router-config
Router (config)# end
Router# copy system:running-config nvram:startup-config
Router#
```

Related Commands

Command	Description
copy system:running-config nvram:startup-config	Saves the environment variable from the running configuration to the startup configuration.
show bootvar	Displays information about the BOOT environment variable.

boot system

To specify the system image that loads at startup, use the **boot system** command. To remove the startup system image specification, use the **no** form of this command.

boot system *filename*

boot system flash [*flash-fs***:**][*partition-number***:**][*filename*]

no boot system [*filename*]

no boot system flash [*flash-fs***:**][*partition-number***:**][*filename*]

Syntax Description

filename	Specifies the configuration filename of the system image to load at system startup.
flash	Boots from internal flash memory.
*flash-fs***:**	(Optional) flash file system containing the system image to load at startup; valid values are **flash:**, **bootflash**, **slot0**, and **slot1**.
*partition-number***:**	(Optional) Number of the flash memory partition that contains the system image to boot, specified by the optional *filename* argument.
filename	(Optional when used with the **boot system flash** command) Case-senstive name of the system image to load at startup.

Defaults

If you configure the switch to boot from a network server but do not specify a system image file with the **boot system** command, the switch uses the configuration register settings to determine the default system image filename. The switch forms the default boot filename by starting with the word *cisco* and then appending the octal equivalent of the boot field number in the configuration register, followed by a hyphen (-) and the processor type name (cisco*nn-cpu*). Refer to the appropriate hardware installation guide for details on the configuration register and default filename. See also the **config-register** or **confreg** command.

Command Modes

Global configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

This command will not work unless you set the **config-register** command properly.

TFTP boot is not supported on the Catalyst 6500 series switches.

If you do not enter the *ip-address* argument, this value defaults to the IP broadcast address of 255.255.255.255.

The colon is required when entering the *flash-fs***:** argument.

If you omit all arguments that follow the **flash** keyword, the system searches the internal flash memory for the first bootable image.

When using the *partition-number***:** argument, if you do not specify a filename, the route processor loads the first valid file in the specified partition of flash memory. This argument is valid only on route processors that can be partitioned.

The *filename* argument is case sensitive. If you do not specify a *filename*, the switch loads the first valid file in the following:

- The specified flash file system
- The specified partition of flash memory
- The default flash file system if you also omitted the *flash-fs***:** argument

Enter several **boot system** commands to provide a fail-safe method for booting your route processor. The route processor stores and executes the **boot system** commands in the order in which you enter them in the configuration file. If you enter multiple boot commands of the same type (for example, if you enter two commands that instruct the route processor to boot from different network servers), the route processor tries them in the order in which they appear in the configuration file. If a **boot system** command entry in the list specifies an invalid device, the route processor omits that entry. Use the **boot system rom** command to specify use of the ROM system image as a backup to other **boot** commands in the configuration.

For some platforms, you must load the boot image before you load the system image. However, on many platforms, the boot image that you specify loads only if the route processor is booting from a network server or if you do not specify the flash file system. If you specify the file system, the route processor boots faster because it does not need to load the boot image first.

For detailed information, refer to the *Cisco IOS Release 12.2 Command Reference*.

Note When you use the **boot system** command, you affect only the running configuration. You must save the BOOT variable settings to your startup configuration to place the information under ROM monitor control and to have the variable function as expected. Use the **copy system:running-config nvram:startup-config** EXEC command to save the variable from your running configuration to your startup configuration.

To view the contents of the BOOT variable, use the **show bootenv** EXEC command.

Examples

This example shows a system filename with the ROM software as a backup:

```
Router(config)# boot system flash config1
Router(config)# boot system rom
```

This example shows how to boot the system image filenamed igs-bpx-l from partition 2 of the flash device:

```
Router(config)# boot system flash:2:igs-bpx-l
Router(config)#
```

Related Commands

Command	Description
config-register	Changes the configuration register settings.
copy /noverify	Disables the automatic image verification for the current copy operation.
ip rcmd remote username	Configures the remote username to be used when requesting a remote copy using rcp.
show bootvar	Displays information about the BOOT environment variable.

bridge-domain

To enable BPDU translation, use the **bridge-domain** command.

bridge-domain {*vlan* | {*PE-vlan* **dot1qtunnel**}} [**ignore-bpdu-pid**] {**pvst-tlv** *CE-vlan*}

Syntax Description

vlan	VLAN number on a back-to back topology.
PE-vlan **dot1qtunnel**	Specifies the provider-edge VLAN number on a Layer 2 topology.
ignore-bpdu-pid	(Optional) Sends out IEEE BPDUs using a PID of 0x00-07, which is normally reserved for RFC 1483 data.
pvst-tlv	When transmitting, translates PVST+ BPDUs into IEEE BPDUs. When receiving, translates IEEE BPDUs into PVST+ BPDUs.
CE-vlan	Customer-edge VLAN in the SSTP TLV to be inserted in an IEEE BPDU to a PVST+ BPDU conversion.

Defaults

Disabled

Command Modes

VC or DLCI configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

The *CE-vlan* argument does not have to be the same as the *PE-vlan* argument.

When connecting to a device that is completely RFC-1483 compliant, in which the IEEE BPDUs are sent using a PID of 0x000E, you must use the **ignore-bpdu-pid** keywords in the **bridge-domain** command.

If you do not enter the **ignore-bpdu-pid** keyword, the PVC between the devices operates in an RFC-1483 compliant topology, which is referred to as *strict mode*. Entering the **ignore-bpdu-pid** keyword enters the *loose mode*. Both modes are described as follows:

- Without the **ignore-bpdu-pid** keywords, in strict mode, IEEE BPDUs are sent out using a PID of 0x00-0E, which complies with RFC 1483.
- With the **ignore-bpdu-pid** keywords, in loose mode, IEEE BPDUs are sent out using a PID of 0x00-07, which is normally reserved for RFC-1483 data.

Cisco-proprietary PVST+ BPDUs are always sent out on data frames using a PID of 0x00-07, regardless of whether you enter the **ignore-bpdu-pid** keywords.

Use the **ignore-bpdu-pid** keywords when connecting to devices (such as ATM DSL modems) that send PVST (or 802.1D) BPDUs with PID: 00-07.

The **pvst-tlv** keyword enables BPDU translation when interoperating with devices that understand only PVST or IEEE Spanning Tree Protocol. Because the Catalyst 6500 series switch ATM modules support PVST+ only, you must use the **pvst-tlv** keyword when connecting to a Catalyst 5000 family switch, which only understands PVST on its ATM modules, or when connecting with other Cisco IOS route processors, which understand IEEE format only.

When transmitting, the **pvst-tlv** keyword translates PVST+ BPDUs into IEEE BPDUs.

When receiving, the **pvst-tlv** keyword translates IEEE BPDUs into PVST+ BPDUs.

Examples

This example shows how to enable BPDU translation when a Catalyst 6500 series switch is connected to a a device that only understand IEEE BPDUs in an RFC-1483 compliant topology:

```
Router(config-if-atm-vc)# bridge-domain 100 pvst-tlv 150
Router(config-if-atm-vc)#
```

The **ignore-bpdu-pid** keyword is not used because the device operates in an RFC-1483 compliant topology for IEEE BPDUs.

This example shows how to enable BPDU translation when a Catalyst 5500 ATM module is a device that only understands PVST BPDUs in a non-RFC1483 compliant topology. When a Catalyst 6500 series switch is connected to a Catalyst 5500 ATM module, you must enter both keywords:

```
Router(config-if-atm-vc)# bridge-domain 100 ignore-bpdu-pid pvst-tlv 150
Router(config-if-atm-vc)#
```

To enable BPDU translation for the Layer 2 Protocol Tunneling (L2PT) topologies, use the following command line:

```
Router(config-if-atm-vc)# bridge-domain 100 dot1qtunnel ignore-bpdu-pid pvst-tlv 150
Router(config-if-atm-vc)#
```

cd

To change the default directory or file system, use the **cd** command.

cd [*filesystem***:**][*directory*]

Syntax Description

*filesystem***:**	(Optional) URL or alias of the directory or file system that is followed by a colon; see the "Usage Guidelines" section for a list of the valid values.
directory	(Optional) Name of the directory.

Defaults

Initial default file system is **disk0:**

Command Modes

EXEC

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

The valid values for *filesystem***:** are **bootflash:**, **disk0:** and **disk1:**.

For all EXEC commands that have an optional *filesystem* argument, the system uses the file system that is specified by the **cd** command when you omit the optional *filesystem* argument. For example, the **dir** command, which displays a list of files on a file system, contains an optional *filesystem* argument. When you omit this argument, the system lists the files on the file system that is specified by the **cd** command.

If you do not specify a directory on a file system, the default is the root directory on that file system.

Examples

This example sets the default file system to the flash PC card that is inserted in disk 0:

```
Router# cd disk0:
Router# pwd
disk0:/
```

Related Commands

Command	Description
dir	Displays a list of files on a file system.
mkdir disk0:	Creates a new directory in a flash file system.
pwd	Displays the current setting of the **cd** command.
show file system	Displays the available file systems.
undelete	Recovers a file that is marked “deleted” on a flash file system.

channel-group

To assign and configure an EtherChannel interface to an EtherChannel group, use the **channel-group** command. To remove the channel-group configuration from the interface, use the **no** form of this command.

channel-group *number* **mode** {**active** | **on** | {**auto** [**non-silent**]} | {**desirable** [**non-silent**]} | **passive**}

no channel-group *number*

Syntax Description

number	Channel-group number; valid values are a maximum of 64 values ranging from 1 to 256.
mode	Specifies the EtherChannel mode of the interface.
active	Enables LACP unconditionally.
on	Enables EtherChannel only.
auto	Places a port into a passive negotiating state in which the port responds to PAgP packets that it receives but does not initiate PAgP packet negotiation.
non-silent	(Optional) Used with the **auto** or **desirable** mode when traffic is expected from the other device.
desirable	Places a port into an active negotiating state in which the port initiates negotiations with other ports by sending PAgP packets.
passive	Enables LACP only if an LACP device is detected.

Defaults

No channel groups are assigned.

Command Modes

Interface configuration

Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

Usage Guidelines

Note You cannot make any changes to the configuration of the Supervisor Engine 32 PISA EtherChannel.

Note After the port becomes a member of the Supervisor Engine 32 PISA EtherChannel, only the **no channel-group 256 mode on** command has any effect on the port until the port is no longer a member of the PISA EtherChannel. While the port is a member of the PISA EtherChannel, all port configuration commands except the **no channel-group 256 mode on** command are ignored.

By default, the Supervisor Engine 32 PISA EtherChannel (port channel interface 256, which is automatically configured with the **pisa-channel** command) is a 1-Gps EtherChannel.

Note The **pisa-channel** command is visible in the configuration file, but it is not user configurable.

The channel-group number is global and is shared between all the channeling protocols. If a specific channel number is used for the PAgP-enabled interfaces of a channel group, that same channel number cannot be used for configuring a channel that has LACP-enabled interfaces or vice versa.

Entering the **auto** or **desirable** keyword enables PAgP on the specified interface; the command will be rejected if it is issued on an LACP-enabled interface.

The **active** and **passive** keywords are valid on PAgP-disabled interfaces only.

You can change the mode for an interface only if it is the only interface that is designated to the specified channel group.

The **on** keyword forces the bundling of the interface on the channel without any negotiation.

You can manually configure a switch with PAgP on one side and LACP on the other side in the **on** mode.

With the **on** mode, a usable EtherChannel exists only when a port group in **on** mode is connected to another port group in **on** mode.

If you enter the **channel group** command on an interface that is added to a channel with a different protocol than the protocol you are entering, the command is rejected.

If the interface belongs to a channel, the **no** form of this command is rejected.

All ports in the same channel group must use the same protocol; you cannot run two protocols on one channel group.

PAgP and LACP are not compatible; both ends of a channel must use the same protocol.

You can change the protocol at any time, but this change causes all existing EtherChannels to reset to the default channel mode for the new protocol.

Configure all ports in an EtherChannel to operate at the same speed and duplex mode (full duplex only for LACP mode).

You do not have to create a port-channel interface before assigning a physical interface to a channel group. A port-channel interface is created automatically when the channel group gets its first physical interface, if it is not already created.

You do not have to disable the IP address that is assigned to a physical interface that is part of a channel group, but it is highly recommended.

You can create both Layer 2 and Layer 3 port channels by entering the **interface port-channel** command or when the channel group gets its first physical interface assignment. The port channels are not created at runtime or dynamically.

Any configuration or attribute changes that you make to the port-channel interface are propagated to all interfaces within the same channel group as the port channel (for example, configuration changes are also propagated to the physical interfaces that are not part of the port channel but are part of the channel group).

When configuring Layer 2 EtherChannels, you cannot put Layer 2 LAN ports into manually created port-channel logical interfaces.

You cannot use the **channel-group** command on GE-WAN interfaces if MPLS is configured. You must remove all IP, MPLS, and other Layer 3 configuration commands before using the **channel-group** command with GE-WAN interfaces.

Note You can enter the **channel-group** command again to delete the interface from the old group and move it to the new group. For GE-WAN ports, however, you must manually remove the interface from the group by entering the **no channel-group** command before assigning it to a new group.

Caution Do not enable Layer 3 addresses on the physical EtherChannel interfaces. Assigning bridge groups on the physical EtherChannel interfaces causes loops in your network.

For a complete list of guidelines, refer to the "Configuring EtherChannel" section of the *Catalyst Supervisor Engine 32 PISA Cisco IOS Software Configuration Guide—Release 12.2ZY*.

Examples

This example shows how to add EtherChannel interface 1/0 to the EtherChannel group that is specified by port-channel 1:

```
Router(config-if)# channel-group 1 mode on
Router(config-if)#
```

Related Commands

Command	Description
interface port-channel	Creates a port-channel virtual interface and enters interface configuration mode.
show etherchannel	Displays the EtherChannel information for a channel.
show interfaces port-channel	Displays the traffic that is seen by a specific port channel.