

I Commands

- identity, on page 4
- ingress-sa, on page 6
- initiator, on page 7
- in-order-guarantee, on page 8
- install, on page 9
- install all, on page 11
- install clock-module, on page 16
- install license, on page 18
- install module bios, on page 19
- install module epld, on page 20
- install module loader, on page 22
- install ssi, on page 23
- interface, on page 25
- interface fc, on page 27
- interface fcip, on page 29
- interface fc-tunnel, on page 32
- interface gigabitethernet, on page 34
- interface ioa, on page 36
- interface iscsi, on page 37
- interface mgmt, on page 39
- interface port-channel, on page 40
- interface sme, on page 42
- interface sme (Cisco SME cluster node configuration submode), on page 43
- interface vsan, on page 45
- intersight connection, on page 46
- intersight proxy, on page 47
- intersight trustpoint, on page 48
- ioa cluster, on page 49
- ioa site-local, on page 50
- ioa-ping, on page 51
- ip access-group, on page 53
- ip access-list, on page 55
- ip address (FCIP profile configuration submode), on page 62

- ip address (interface configuration), on page 63
- ip default-gateway, on page 64
- ip default-network, on page 65
- ip (destination-group), on page 66
- ip domain-list, on page 68
- ip domain-lookup, on page 69
- ip domain-name, on page 70
- ip name-server, on page 71
- ip route, on page 72
- ip routing, on page 73
- ip-compression, on page 74
- ips netsim delay-ms, on page 76
- ips netsim delay-us, on page 77
- ips netsim drop nth, on page 78
- ips netsim drop random, on page 80
- ips netsim enable, on page 82
- ips netsim max-bandwidth-kbps, on page 83
- ips netsim max-bandwidth-mbps, on page 84
- ips netsim qsize, on page 85
- ips netsim reorder, on page 86
- ipv6 access-list, on page 88
- ipv6 address, on page 89
- ipv6 enable, on page 90
- ipv6 nd, on page 91
- ipv6 route, on page 93
- ipv6 routing, on page 95
- ipv6 traffic-filter, on page 96
- iscsi authentication, on page 97
- iscsi duplicate-wwn-check, on page 99
- iscsi dynamic initiator, on page 101
- iscsi enable, on page 103
- iscsi enable module, on page 104
- iscsi import target fc, on page 105
- iscsi initiator idle-timeout, on page 106
- iscsi initiator ip-address, on page 107
- iscsi initiator name, on page 109
- iscsi interface vsan-membership, on page 110
- iscsi save-initiator, on page 111
- iscsi virtual-target name, on page 113
- islb abort, on page 116
- islb commit, on page 117
- islb distribute, on page 118
- islb initiator, on page 120
- islb save-initiator, on page 122
- islb virtual-target name, on page 124
- islb vrrp, on page 126

- islb zoneset activate, on page 128
- isns, on page 129
- isns distribute, on page 130
- isns esi retries, on page 131
- isns profile name, on page 132
- isns reregister, on page 133
- isns-server enable, on page 134
- ivr aam pre-deregister-check, on page 135
- ivr aam register, on page 136
- ivr abort, on page 137
- ivr commit, on page 138
- ivr copy active-service-group user-configured-service-group, on page 139
- ivr copy active-topology user-configured-topology, on page 140
- ivr copy active-zoneset full-zoneset, on page 141
- ivr copy auto-topology user-configured-topology, on page 142
- ivr distribute, on page 143
- ivr enable, on page 144
- ivr fcdomain database autonomous-fabric-num, on page 145
- ivr nat, on page 146
- ivr refresh, on page 147
- ivr service-group activate, on page 148
- ivr service-group name, on page 149
- ivr virtual-fcdomain-add, on page 151
- ivr virtual-fcdomain-add2, on page 152
- ivr vsan-topology, on page 153
- ivr vsan-topology auto, on page 155
- ivr vsan-topology database, on page 156
- ivr withdraw domain, on page 158
- ivr zone name, on page 159
- ivr zone rename, on page 160
- ivr zoneset, on page 161
- ivr zoneset rename, on page 162

identity

To configure the identity for the IKE protocol, use the **identity** command in IKE configuration submode. To delete the identity, use the **no** form of the command.

identity {**address** | **hostname**}
no identity {**address** | **hostname**}

Syntax Description

address	Sets the IKE identity to be the IPv4 address of the switch.
hostname	Sets the IKE identity to be the host name of the switch.

Command Default

None.

Command Modes

IKE configuration submode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

To use this command, the IKE protocol must be enabled using the **crypto ike enable** command.

Before configuring a certificate for the switch, configure the host name and domain name, and set the identity to be the host name. This allows the certificate to be used for authentication.

Note The host name is the fully qualified domain name (FQDN) of the switch. To use the switch FQDN for the IKE identity, you must first configure both the switch name and the domain name. The FQDN is required for using RSA signatures for authentication. By default address is identified.

Examples

The following example shows how to set the IKE identity to the IP address of the switch:

```
switch# config terminal
switch(config)# crypto ike domain ipsec
switch(config-ike-ipsec)# identity address
```

The following example shows how to delete the IKE identity:

```
switch(config-ike-ipsec)# no identity address
```

The following example shows how to set the IKE identity to the host name:

```
switch(config-ike-ipsec)# identity hostname
```

The following example shows how to delete the IKE identity:

```
switch(config-ike-ipsec)# no identity hostname
```

Related Commands

Command	Description
crypto ike domain ipsec	Enters IKE configuration mode.
crypto ike enable	Enables the IKE protocol.
show crypto ike domain ipsec	Displays IKE information for the IPsec domain.

ingress-sa

To configure the Security Association (SA) to the ingress hardware, use the **ingress-sa** command. To delete the SA from the ingress hardware, use the **no** form of the command.

ingress-sa *spi-number*
no ingress-sa *spi-number*

Syntax Description

spi-number	The range is from 256 to 4294967295.

Command Default

None.

Command Modes

Configuration submode.

Command History

Release	Modification
NX-OS 4.2(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to configure the SA to the ingress hardware:

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# interface fc 2/1 - 3
switch(config-if)# fcsp esp manual
switch(config-if-esp)# ingress-sa 258
switch(config-if-esp)#
```

Related Commands

Command	Description
show fcsp interface	Displays FC-SP-related information for a specific interface.

initiator

To configure the initiator version and address, use the **initiator** command IKE configuration submode. To revert to the default, use the **no** form of the command.

initiator version *version* **address** *ip-address*
no initiator version *version* **address** *ip-address*

Syntax Description

version	Specifies the protocol version number. The only valid value is 1.
address *ip-address*	Specifies the IP address for the IKE peer. The format is *A.B.C.D*.

Command Default

IKE version 2.

Command Modes

IKE configuration submode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

To use this command, the IKE protocol must be enabled using the **crypto ike enable** command.

Examples

The following example shows how initiator information for the IKE protocol:

```
switch# config terminal
switch(config)# crypto ike domain ipsec
switch(config-ike-ipsec)# initiator version 1 address 10.1.1.1
```

Related Commands

Command	Description
crypto ike domain ipsec	Enters IKE configuration mode.
crypto ike enable	Enables the IKE protocol.
show crypto ike domain ipsec	Displays IKE information for the IPsec domain.

in-order-guarantee

To enable in-order delivery, use the **in-order-guarantee** command in configuration mode. To disable in-order delivery, use the **no** form of the command.

in-order-guarantee [**vsan** *vsan-id*]
no in-order-guarantee [**vsan** *vsan-id*]

Syntax Description

vsan *vsan-id*	(Optional) Specifies a VSAN ID. The range is 1 to 4093.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(4)	This command was introduced.

Usage Guidelines

In-order delivery of data frames guarantees frame delivery to a destination in the same order that they were sent by the originator.

Examples

The following example shows how to enable in-order delivery for the entire switch:

```
switch# config terminal
switch(config) # in-order-guarantee
```

The following example shows how to disable in-order delivery for the entire switch:

```
switch(config)# no in-order-guarantee
```

The following example shows how to enable in-order delivery for a specific VSAN:

```
switch(config)# in-order-guarantee vsan 3452
```

The following example shows how to disable in-order delivery for a specific VSAN:

```
switch(config)# no in-order-guarantee vsan 101
```

Related Commands

Command	Description
show in-order-guarantee	Displays the in-order-guarantee status.

install

To upgrade hardware firmware on individual switch components, use the **install** command.

install fan-module *fan_number* **epld** *uri*
install module *module_number* **firmware** *uri*
install transceiver { *uri* **module** *mod_range* | **clean** }
install xbar *xbar_number* **epld** *uri*

install	Upgrade system
clean	Clean up aborted transceiver update.
epld	Install firmware on EPLDs.
fan-module	Install on a fan module.
fan_number	Fan module number.
firmware	Install firmware on linecard EPLD.
module	Specify module number.
mod_range	Module number or range.
transceiver	Install on transceivers.
xbar	Install on a fabric module.
xbar_number	Fabric module number.
uri	URI of package containing EPLD.

Command Default

None

Command Modes

EXEC mode

Command History

Release	Modification
1.0 (3)	This command was added

Usage Guidelines

Most EPLD upgrades require the component to be reloaded after the upgrade to begin using the installed firmware. EPLD firmware is not part of the NX-OS software image. Firmware for a platform is a seperate bundle of EPLD images. For more information about how to select and install EPLD and transceiver updates, refer to the EPLD and transceiver release notes and installation guides.

Example

To install firmware updates on

```
switch# install fan-module 1 epld bootflash:mds9000-s1-epld.9.4.1.img

EPLD                                 Curr Ver    New Ver
-----------------------------------------------------------
Fan Controller                        0.005       0.007

WARNING: Upgrade process could take up to 30 minutes. Module could be powered down and up.
 Programming Fan Module 1
Do you want to continue (y/n) ? [n] y

Programming Fan Module 1 EPLD...
Fan Module 1 EPLD upgrade successful.
```

Related Commands

Table 1: Related Commands

Command Syntax	Description
show hardware internal version	Displays installed EPLD versions.
show interface transceiver	Displays transceiver firmware version.

install all

To upgrade all modules in any Cisco MDS 9000 family switch, use the **install all** command.

install all kickstart *ks_uri* **system** *system_uri*
install all epld *epld_uri* **parallel fan** { *fan_number* | **all }**
install all epld *epld_uri* **parallel xbar** { *xbar_range* | **all }**
install all epld *epld_uri* **parallel mod** { *mod_range* | **all }**

Syntax Description

install	Upgrade system.
all	Upgrade all modules.
kickstart	Specify the kickstart image.
ks_uri	URI of the kickstart image.
system	Specify the system image.
system_uri	URI of the system image.
epld	Install firmware on EPLDs.
parallel	Install on all modules simultaneously
fan-module	Install on a fan module.
fan-range	Fan module number range.
module	Specify module number.
mod_range	Module number or range.
xbar	Install on a fabric module.
xbar_range	Fabric module number range.
epld_uri	URI of package containing EPLD.

The following table lists the supported options for image URIs. Not all command options support all URI options.

bootflash:	Source location for internal bootflash memory.
slot0:	Source location for the CompactFlash memory or PCMCIA card.
volatile:	Source location for the volatile file system.
tftp:	Source location for a Trivial File Transfer Protocol (TFTP) network server. The syntax for this URL is **tftp:**[[//*server_addr*]/*directory*]/*filename*
ftp:	Source location for a File Transfer Protocol (FTP) network server. The syntax for this URL is **ftp:**[[//*server_addr*]/*directory*]/*filename*

sftp:	Source location for a Secure Trivial File Transfer Protocol (SFTP) network server. The syntax for this URL is **sftp:**[[//[*username*@]*server_addr*]/*directory*]/*filename*
scp:	Source location for a Secure Copy Protocol (SCP) network server. The syntax for this URL is **scp:**[[//[*username*@]*server_addr*]/*directory*]/*filename*.
filename	The name of the source image file.

Command Default

None.

Command Modes

Exec mode.

Command History

Release	Modification
1.0(3)	This command was introduced.
1.2(2)	Added the **asm-sfn** keyword and made all keywords optional.
2.0(1b)	Added the **ssi** keyword.

Usage Guidelines

A system upgrade is nondisruptive except when certain components are upgraded. Upgrades of EPLDs and FCIP engines are disruptive.

Most EPLD upgrades require the component to be reloaded after the upgrade to begin using the installed firmware. EPLD firmware is not part of the NX-OS software image. Firmware for a platform is a seperate bundle of EPLD images. For more information about how to select and install EPLD and transceiver updates, refer to the EPLD and transceiver release notes and installation guides.

A system software upgrade using the **install all kickstart system** command supports installation from images on remote servers. It may be necessary to enter credentials, depending on the protocol. The images are copied to bootflash so enough free space for any remotely copied images is required for the install to succeed.



Caution If a switchover is required when you issue the **install all** command from a Telnet or SSH session, all open sessions are terminated. If no switchover is required, the session remains connected.

Example

Examples

The following example shows a system software upgrade with the **install all** command if the system and kickstart files are stored locally:

```
switch# install all sys bootflash:isan-1.3.1 kickstart bootflash:boot-1.3.1

Verifying image bootflash:/boot-1.3.1
[####################] 100% -- SUCCESS

Verifying image bootflash:/isan-1.3.1
[####################] 100% -- SUCCESS

Extracting "slc" version from image bootflash:/isan-1.3.1.
```

```
[####################] 100% -- SUCCESS

Extracting "ips" version from image bootflash:/isan-1.3.1.
[####################] 100% -- SUCCESS

Extracting "system" version from image bootflash:/isan-1.3.1.
[####################] 100% -- SUCCESS

Extracting "kickstart" version from image bootflash:/boot-1.3.1.
[####################] 100% -- SUCCESS

Extracting "loader" version from image bootflash:/boot-1.3.1.
[####################] 100% -- SUCCESS


Compatibility check is done:
Module  bootable          Impact  Install-type  Reason
------  --------  --------------  ------------  ------
     1       yes  non-disruptive       rolling
     2       yes      disruptive       rolling  Hitless upgrade is not supported
     3       yes      disruptive       rolling  Hitless upgrade is not supported
     4       yes  non-disruptive       rolling
     5       yes  non-disruptive         reset
     6       yes  non-disruptive         reset


Images will be upgraded according to following table:
Module       Image       Running-Version           New-Version  Upg-Required
------  ----------  --------------------  --------------------  ------------
     1         slc               1.3(2a)               1.3(1)           yes
     1        bios      v1.1.0(10/24/03)      v1.1.0(10/24/03)            no
     2         ips               1.3(2a)               1.3(1)           yes
     2        bios      v1.1.0(10/24/03)      v1.1.0(10/24/03)            no
     3         ips               1.3(2a)               1.3(1)           yes
     3        bios      v1.1.0(10/24/03)      v1.1.0(10/24/03)            no
     4         slc               1.3(2a)               1.3(1)           yes
     4        bios      v1.1.0(10/24/03)      v1.1.0(10/24/03)            no
     5      system               1.3(2a)               1.3(1)           yes
     5   kickstart               1.3(2a)               1.3(1)           yes
     5        bios      v1.1.0(10/24/03)      v1.1.0(10/24/03)            no
     5      loader                1.2(2)               1.2(2)            no
     6      system               1.3(2a)               1.3(1)           yes
     6   kickstart               1.3(2a)               1.3(1)           yes
     6        bios      v1.1.0(10/24/03)      v1.1.0(10/24/03)            no
     6      loader                1.2(2)               1.2(2)            no


Do you want to continue with the installation (y/n)?  [n] y

Install is in progress, please wait.

Syncing image bootflash:/boot-1.3.1 to standby.
[####################] 100% -- SUCCESS

Syncing image bootflash:/isan-1.3.1 to standby.
[####################] 100% -- SUCCESS

Jan 18 23:40:03 Hacienda %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
Performing configuration copy.
[####################] 100% -- SUCCESS

Module 6: Waiting for module online.
|
Auto booting bootflash:/boot-1.3.1 bootflash:/isan-1.3.1...
```

```
Booting kickstart image: bootflash:/boot-1.3.1....
.....................................Image verification OK

Starting kernel...
INIT: version 2.78 booting
Checking all filesystems..r.r.. done.
Loading system software
Uncompressing system image: bootflash:/isan-1.3.1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
INIT: Entering runlevel: 3
```

The following example shows a system software upgrade of the **install all** command if the system and kickstart files are stored remotely:

```
switch# install all system scp://user@171.69.16.26/tftpboot/m9500-sf1ek9-mz.1.3.2a.bin
kickstart
scp://user@171.69.16.26/tftpboot/HKrel/qa/vegas/final/m9500-sf1ek9-kickstart-mz.1.3.2a.bin
For scp://user@171.69.16.26, please enter password:
For scp://user@171.69.16.26, please enter password:

Copying image from
scp://user@171.69.16.26/tftpboot/m9500-sf1ek9-kickstart-mz.1.3.2a.bin to
bootflash:///m9500-sf1ek9-kickstart-mz.1.3.2a.bin.
[####################] 100% -- SUCCESS

Copying image from
scp://user@171.69.16.26/tftpboot/m9500-sf1ek9-mz.1.3.2a.bin to
bootflash:///m9500-sf1ek9-mz.1.3.2a.bin.
[####################] 100% -- SUCCESS

Verifying image bootflash:///m9500-sf1ek9-kickstart-mz.1.3.2a.bin
[####################] 100% -- SUCCESS

Verifying image bootflash:///m9500-sf1ek9-mz.1.3.2a.bin
[####################] 100% -- SUCCESS

Extracting “slc” version from image bootflash:///m9500-sf1ek9-mz.1.3.2a.bin.
[####################] 100% -- SUCCESS

Extracting “ips” version from image bootflash:///m9500-sf1ek9-mz.1.3.2a.bin.
[####################] 100% -- SUCCESS

Extracting “system” version from image bootflash:///m9500-sf1ek9-mz.1.3.2a.bin.
[####################] 100% -- SUCCESS

Extracting “kickstart” version from image bootflash:///m9500-sf1ek9-kickstart-mz.1.3.2a.bin.
[####################] 100% -- SUCCESS

Extracting “loader” version from image bootflash:///m9500-sf1ek9-kickstart-mz.1.3.2a.bin.
[####################] 100% -- SUCCESS

Compatibility check is done:
Module  bootable          Impact  Install-type  Reason
------  --------  --------------  ------------  ------
     1       yes  non-disruptive       rolling
     2       yes      disruptive       rolling  Hitless upgrade is not supported
     3       yes  non-disruptive       rolling
     4       yes  non-disruptive       rolling
     5       yes  non-disruptive         reset
     6       yes  non-disruptive         reset
     7       yes  non-disruptive       rolling
     8       yes  non-disruptive       rolling
```

```
     9       yes      disruptive       rolling  Hitless upgrade is not supported
...
```

Related Commands

Command	Description
show installallstatus	Displays the progress of a software upgrade.
show version	Displays currently running software version information.

install clock-module

To upgrade the EPLD images of the clock module on a Cisco MDS 9513 Switch Director, use the **install clock-module** command.

install clock-module [**epld** {**bootflash:** | **slot0:** | **volatile:**}]

Syntax Description

epld	(Optional) Installs the clock module EPLD from the EPLD image.
bootflash:	(Optional) Specifies the local URI containing EPLD image.
slot0:	(Optional) Specifies the local URI containing EPLD image.
volatile:	(Optional) Specifies the local URI containing EPLD image.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Use this command on the active supervisor to install the standby clock module EPLD from the specified EPLD image. After upgrading the clock module, power cycle the entire chassis for the change to take effect. It is not sufficient to reboot the chassis; you must turn the power off and on.

Note This command is supported only on the Cisco MDS 9513 Multilayer Switch Director.

Examples

The following example upgrades the EPLD images for the clock module:

```
switch# install clock-module epld bootflash:m9000-epld-3.0.0.278.img
Len 3031343, CS 0x58, string MDS series EPLD image, built on Fri Nov 11 01:11:09 2005
EPLD Curr Ver New Ver
------------------------------------------------------
Clock Controller 0x03 0x04
There are some newer versions of EPLDs in the image!
Do you want to continue (y/n) ? y
Proceeding to program Clock Module B.
Do you want to switchover Clock Modules after programming Clock Module B.
System Will Reset! y/n) ?n

|


Clock Module B EPLD upgrade is successful.
```

Related Commands

Command	Description
show version clock-module epld	Displays the current EPLD versions on the clock module.

install license

To program the supervisor or switching module BIOS, use the **install license** command.

install license [**bootflash:** | **slot0:** | **volatile:**] *file-name*

Syntax Description

bootflash:	(Optional) Specifies the source location for the license file.
slot0:	(Optional) Specifies the source location for the license file.
volatile:	(Optional) Specifies the source location for the license file.
file-name	Specifies the name of the license file.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
1.2(1)	This command was introduced.
9.2(2) and later releases	This command was deprecated.

Usage Guidelines

If a target filename is provided after the source URL, the license file is installed with that name. Otherwise, the filename in the source URL is used. This command also verifies the license file before installing it.

Examples

The following example installs a file named license-file which resides in the bootflash: directory:

```
switch# install license bootflash:license-file
```

Related Commands

Command	Description
show license	Displays license information.

install module bios

To program the supervisor or switching module BIOS, use the **install module bios** command.

install module *module-number* **bios** {**system** [**bootflash:** | **slot0:** | **volatile:** *system-image*]}

Syntax Description

module-number	Specifies the module number from slot 1 to 9 in a Cisco MDS 9500 Series switch. Specifies the module number from slot 1 to 2 in a Cisco MDS 9200 Series switch.
system	(Optional) Specifies the system image to use (optional). If system is not specified, the current running image is used.
bootflash:	(Optional) Specifies the source location for internal bootflash memory
slot0:	(Optional) Specifies the source location for the CompactFlash memory or PCMCIA card.
volatile:	(Optional) Specifies the source location for the volatile file system.
system-image	(Optional) Specifies the name of the system or kickstart image.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
1.0(3)	This command was introduced.

Usage Guidelines

If the BIOS is upgraded, you need to reboot to make the new BIOS effective. You can schedule the reboot at a convenient time so traffic will not be impacted.

The console baud rate automatically reverts to the default rate (9600) after any BIOS upgrade.

The URL is always the system image URL in the supervisor module, and points to the bootflash: or slot0: directories.

Examples

The following example shows how to perform a nondisruptive upgrade for the system:

```
switch# install module 1 bios
Started bios programming .... please wait
###
BIOS upgrade succeeded for module 1
```

In this example, the switching module in slot 1 was updated.

install module epld

To upgrade the electrically programmable logical devices (EPLDs) module, use the **install module epld** command. This command is only for supervisor modules, not switching modules.

install module *module-number* **epld** [**bootflash:** | **ftp:** | **scp:** | **sftp:** | **tftp:** | **volatile:**]

Syntax Description

module-number	Enters the number for the standby supervisor modules or any other line card.
bootflash:	(Optional) Specifies the source location for internal bootflash memory.
ftp	(Optional) Specifies the local/remote URI containing EPLD image.
scp	(Optional) Specifies the local/remote URI containing EPLD image.
sftp	(Optional) Specifies the local/remote URI containing EPLD image.
tftp	(Optional) Specifies the local/remote URI containing EPLD image.
volatile:	(Optional) Specifies the source location for the volatile file system.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
1.2(1)	This command was introduced.

Usage Guidelines

Issue this command from the active supervisor module to update any other module.

If you forcefully upgrade a module that is not online, all EPLDs are forcefully upgraded. If the module is not present in the switch, an error is returned. If the module is present, the command process continues.

Do not insert or extract any modules while an EPLD upgrade or downgrade is in progress.

Examples

The following example upgrades the EPLDs for the module in slot 2:

```
switch# install module 2 epld scp://user@10.6.16.22/users/dino/epld.img

The authenticity of host '10.6.16.22' can't be established.
RSA1 key fingerprint is 55:2e:1f:0b:18:76:24:02:c2:3b:62:dc:9b:6b:7f:b7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.6.16.22' (RSA1) to the list of known hosts.
user@10.6.16.22's password:
epld.img             100% |****************************|  1269 KB    00:00
Module Number                                       2
EPLD                              Curr Ver     New Ver
-----------------------------------------------------
Power Manager                        0x06
XBUS IO                              0x07        0x08
```

```
UD chip Fix                          0x05
Sahara                               0x05        0x05

Module 2 will be powered down now!!
Do you want to continue (y/n) ? y
\ <-----------------------------------------------------------progress twirl
Module 2 EPLD upgrade is successful
```

The following example forcefully upgrades the EPLDs for the module in slot 2:

```
switch# install module 2 epld scp://user@10.6.16.22/epld-img-file-path

Module 2 is not online, Do you want to continue (y/n) ? y
cchetty@171.69.16.22's password:
epld.img             100% |****************************|  1269 KB    00:00
\ <-----------------------------------------------------------progress twirl
Module 2 EPLD upgrade is successful
```

Related Commands

Command	Description
show version epld	Displays the available EPLD versions.
show version module*number* **epld**	Displays the current EPLD versions.

install module loader

To upgrade the bootloader on either the active or standby supervisor module, use the **install module loader** command. This command is only for supervisor modules, not switching modules.

install module *module-number* **loader kickstart** [**bootflash:** | **slot0:** | **volatile:** *kickstart-image*]

Syntax Description

module-number	Enters the module number for the active or standby supervisor modules (only slot 5 or 6).
kickstart	Specifies the kickstart image to use.
bootflash:	(Optional) Specifies the source location for internal bootflash memory
slot0:	(Optional) Specifies the source location for the CompactFlash memory or PCMCIA card.
volatile:	(Optional) Specifies the source location for the volatile file system.
kickstart-image	Specifies the name of the kickstart image.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
1.0(3)	This command was introduced.

Usage Guidelines

Before issuing the **install module loader** command, be sure to read the release notes to verify compatibility issues between the boot loader and the kickstart or system images.

If you install a loader version that is the same as the currently installed version, the loader will not be upgraded. When both the current version and the installed version are the same, use the **init system** command to force a loader upgrade.

Examples

The following example shows how to perform a non disruptive upgrade for the system:

```
switch# install module 6 loader bootflash:kickstart_image
```

Related Commands

Command	Description
show version	Verifies the output before and after the upgrade.

install ssi

To perform a nondisruptive upgrade of the SSI image on an SSM, use the **install ssi** command.

install ssi {**bootflash:** | **slot0:** | **modflash:**} *file-name* **module** *slot*

Syntax Description

bootflash:	Specifies the source location for the SSI boot image file.
slot0:	Specifies the source location for the SSI boot image file.
modflash:	Specifies the source location for the SSI boot image file.
file-name	Specifies the SSI boot image filename.
module *slot*	Specifies the module slot number.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
5.0(x)	This command has been deprecated (install ssi command is not supported for gen 2 card.
2.1(2)	This command was introduced.

Usage Guidelines

You can use the **install ssi** command to upgrade or downgrade the SSI boot image if the SSM is only configured for Fibre Channel switching. If your SSM is configured for VSFN or Intelligent Storage Services, you must use the **boot** command to reconfigure the SSI boot variable and reload the module.

The **install ssi** command implicitly sets the SSI boot variable.

Note The SSM must be running EPLD version 2.1(2) to use the **install ssi** command. You must install the SSM on a Cisco MDS 9500 Series switch to update the EPLD.

Note The **install ssi** command does not support files located on the SSM modflash.

Examples

The following example installs the SSI boot image on the module in slot 2:

```
switch# install ssi bootflash:lm9000-ek9-ssi-mz.2.1.2.bin module 2
```

Related Commands

Command	Description
boot	Configures the boot variables.
show boot	Displays the current contents of boot variables.
show module	Verifies the status of a module.

interface

To configure an interface on the Cisco MDS 9000 Family of switches, use the **interface** command in configuration mode.

interface {**cpp** {*module-numberprocessor-numbervsan-id*} | **ethernet** {*slot number\ port-number*} | **ethernet-port-channel** *ethernet-port-channel-number* | **fc** {*slot number* | **port number** | **fc-tunnel** *tunnel-id*} | **mgmt** | **port-channel** *port-channel-number* | **vfc** *vfc-id* | **vfc port-channel** *vfc port-channel-id* | **vsan** *vsan-id*}
nointerface {**cpp** {*module-numberprocessor-numbervsan-id*} | **ethernet** {*slot number\ port-number*} | **ethernet-port-channel** *ethernet-port-channel-number* | **fc** {*slot number* | **port number** | **fc-tunnel** *tunnel-id*} | **mgmt** | **port-channel** *port-channel-number* | **vfc** *vfc-id* | **vfc port-channel** *vfc port-channel-id* | **vsan** *vsan-id*}



Note On a Cisco Fabric Switch for HP c-Class BladeSystem and on a Cisco Fabric Switch for IBM BladeCenter, the syntax differs as follows:

Syntax Description

cpp	Configures a Control Plane Process (CPP) interface.
module-number	Specifies the module number. The range is 1 to 10.
processor-number	Specfies the processor number. The range is from 1 to 1.
vsan-id	Specifies the VSAN ID. The range is from 1 to 4093.
ethernet	Specfies the Ethernet IEEE 802.3z.
slot number / *port number*	Specifies the Ethernet slot number and port number. Slot range is from 1 to 253 and port number range is from 1 to 128.
ethernet-port-channel	Ethernet Port Channel interface. The range is from 513 to 4096.
ethernet-port-channel-number	Specifies the Port Channel number. The range is from 513 to 4096.
fc	(Optional) Configures a Fiber Channel interface on an MDS 9000 Family switch (see the **interface fc** command).
slot number / *port number*	Specifies the slot number. The range is from 1 to 10. Specifies the FC slot number and port number. Slot range is from 1 to 10 and port number range is from 1 to 48.
fc-tunnel	Configures a Fiber Channel link interface (see the **interface fc-tunnel** command).
tunnel-id	Specifies the tunnel ID. The range is from 1 to 255.
mgmt	Configures a management interface (see the **interface mgmt** command).

port-channel	Configures a Port Channel interface (see the **interface port-channel** command).
port-channel-number	Specifies the Port Channel number. The range is from 1 to 256.
vfc	Specifies the Virtual FC interface.
vfc-id	Specifies the virtual interface ID or slot. The range is from 1 to 8192.
vfc-port-channel	Specifies the virtual FC port-channel interface
vfc-port-channel-id	Specifies the virtual interface ID. The range is from 513 to 4096.
vsan	Specifies the IPFC VSAN interface.
vsan-id	Specfies the VSAN ID. The range is from 1 to 4093.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(2)	This command was introduced.

Usage Guidelines

You can specify a range of interfaces by issuing a command with the following example format:

interface fc1/1 - 5 , fc2/5 - 7

The spaces are required before and after the dash (-) and before and after the comma (,).



Note For Cisco MDS 9500, 9700 and 9250i Series Switches support ethernet , vfc, vfc-port-channel and ethernet-port-channel commands.

Examples

The following example selects the mgmt 0 interface and enters interface configuration submode:

```
switch# config terminal
switch(config)# interface mgmt 0
```

Related Commands

Command	Description
show interface	Displays an interface configuration for a specified interface.

interface fc

To configure a Fibre Channel interface on the Cisco MDS 9000 Family of switches, use the **interface fc** command in EXEC mode. To revert to defaults, use the **no** form of the command.

interface fc *slot/port* **channel-group** {*group-id* [**force**] | **auto**} **fcdomain rcf-reject vsan** *vsan-id* **fcsp**
| fspf {**cost** *link-cost* **vsan** *vsan-id* | **ficon portnumber** *portnumber* | **dead-interval** *seconds* **vsan** *vsan-id* | **hello-interval** *seconds* **vsan** *vsan-id* | **passive vsan** *vsan-id* | **retransmit-interval** *seconds* **vsan** *vsan-id*}
nointerface fc *slot/port* **channel-group** {*group-id* [**force**] | **auto**} **fcdomain rcf-reject vsan** *vsan-id* **no fspf** {**cost** *link-cost* **vsan** *vsan-id* | **ficon portnumber** *portnumber* | **dead-interval** *seconds* **vsan** *vsan-id* | **hello-interval** *seconds* **vsan** *vsan-id* | **passive vsan** *vsan-id* | **retransmit-interval** *seconds* **vsan** *vsan-id*}

Syntax Description

slot/port	Specifies a slot number and port number.
channel-group	Add to or remove chaneel group from a Port Channel.
group-id	Specifies a Port Channel group number from 1 to 128.
force	(Optional) Forcefully adds a port.
auto	Enables autocreation of Port Channels.
fcdomain	Enters the interface submode.
rcf-reject	Configures the rcf-reject flag.
vsan *vsan-id*	Specifies the VSAN ID. The range is 1 to 4093.
fcsp	Configures the FCSP for an interface.
fspf	Configures FSPF parameters.
cost *link-cost*	Configures FSPF link cost. The range is 1 to 30000.
ficon	Configures FICON parameters.
portnumber *portnumber*	Configures the FICON port number for this interface.
dead-interval *seconds*	Configures FSPF dead interval in seconds. The range is 2 to 65535.
hello-interval *seconds*	Configures FSPF hello-interval. The range is 1 to 65535.
passive	Enables or disables FSPF on the interface.
retransmit-interval *seconds*	Configures FSPF retransmit interface in seconds. The range is 1 to 65535.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
NX-OS 4.2(1)	Added fcsp keyword for the syntax description.
1.0(2)	This command was introduced.
2.0(x)	Added the **auto** option to the **channel-group** keyword.

Usage Guidelines

You can specify a range of interfaces by entering the command with the following example format:

interface*space***fc1/1***space-space***5***space*,*space***fc2/5***space-space***7**

Use the **no shutdown** command to enable the interface.

The **channel-group auto** command enables autocreation of Port Channels. If autocreation of Port Channels is enabled for an interface, you must first disable this configuration before downgrading to earlier software versions or before configuring the interface in a manually configured channel group.

Examples

The following example configures ports 1 to 4 in Fibre Channel interface 9:

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# int fc9/1 - 4
```

The following example enables the Fibre Channel interface:

```
switch# config terminal
switch(config)# interface fc1/1
switch(config-if)# no shutdown
```

The following example assigns the FICON port number to the selected Fibre Channel interface:

```
switch# config terminal
switch(config)# interface fc1/1
switch(config-if)# ficon portnumber 15
```

Related Commands

Command	Description
show interface	Displays an interface configuration for a specified interface.
shutdown	Disables and enables an interface.

interface fcip

To configure a Fibre Channel over IP Protocol (FCIP) interface, use the **interface fcip** command. To disable a FCIP interface, use the **no** form of the command.

interface fcip *interface_number* **bport bport-keepalives channel-group** *number* [**force**] **fcdomain rcf-reject vsan** *vsan-id* **ficon portnumber** *portnumber*| **fspf** {**cost** *link-cost* | **dead-interval** *seconds* | **hello-interval** *seconds* | **passive** | **retransmit-interval** *seconds*} **vsan** *vsan-id* **passive-mode peer-info ipaddr** *ip-address* [**port** *number*] **qos control** *control-value* **data** *data-value* **special-frame peer-wwn** *pwwn-id* **tcp-connections** *number* **time-stamp** [**acceptable-diff** *number*] **use-profile** *profile-id*
no interface fcip *interface_number* **bport bport-keepalives channel-group** *number* [**force**] **fcdomain rcf-reject vsan** *vsan-id* **ficon portnumber** *portnumber* **fspf** {**cost** *link-cost* | **dead-interval** *seconds* | **hello-interval** *seconds* | **passive** | **retransmit-interval** *seconds*} **vsan** *vsan-id* **qos** *control-value* **data** *data-value* **passive-mode peer-info ipaddr** *ip-address* [**port** *number*] **special-frame peer-wwn** *pwwn-id* **tcp-connections** *number* **time-stamp** [**acceptable-diff** *number*] **use-profile** *profile-id*

Syntax Description

interface-number	Configures the specified interface from 1 to 255.
bport	Sets the B port mode.
bport-keepalives	Sets the B port keepalive responses.
channel-group *number*	Specifies a PortChannel number from 1 to 128.
force	(Optional) Forcefully adds a port.
fcdomain	Enters the fcdomain mode for this FCIP interface
rcf-reject	Configures the rcf-reject flag.
vsan *vsan-id*	Specifies a VSAN ID. The range is 1 to 4093.
ficon	Configures FICON parameters.
portnumber *portnumber*	Configures the FICON port number for this interface.
fspf	Configures FSPF parameters.
cost *link-cost*	Enters FSPF link cost. The range is 1 to 30000.
dead-interval *seconds*	Specifies the dead interval in seconds. The range is 1 to 65535.
hello-interval *seconds*	Specifies FSPF hello-interval in seconds. The range is 1 to 65535.
passive	Enables or disables FSPF on the interface.
retransmit-interval	Specifies FSPF retransmit interface in seconds. The range is 1 to 65535.
passive-mode	Configures a passive connection.
peer-info	Configures the peer information.
ipaddr *ip-address*	Specifies the peer IP address.

port *number*	(Optional) Specifies the peer port number. The range is 1 to 65535.
qos	Configures the differentiated services code point (DSCP) value to mark all IP packets.
control *control-value*	Specifies the control value for DSCP.
data *data-value*	Specifies the data value for DSCP.
special-frame	Configures special frames.
peer-wwn *pwwn-id*	Specifies the peer WWN for special frames.
switchport	Configures switchport parameters.
tcp-connections *number*	Specifies the number of TCP connection attempts. Valid values are 1 or 2.
time-stamp	Configures the time stamp.
acceptable-diff *number*	(Optional) Specifies the acceptable time difference for time stamps. The range is 1 to 60000.
use-profile *profile-id*	Specifies the interface using an existing profile ID. The range is 1 to 255.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.1(1)	This command was introduced.
1.3(1)	Added the **ficon portnumber** subcommand.
2.0(x)	Added the **qos** subcommand.

Usage Guidelines

You can specify a range of interfaces by issuing a command with the following example format:

interface **fcip1***space-space***5***space*,*space***fcip10***space-space***12***space*

Examples

The following example selects an FCIP interface and enters interface configuration submode:

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# interface fcip 1
switch(config-if)#
```

The following example assigns the FICON port number to the selected FCIP interface:

```
switch# config terminal
switch(config)# interface fcip 51
switch(config-if)# ficon portnumber 234
```

Related Commands

Command	Description
show interface fcip	Displays an interface configuration for a specified FCIP interface.

interface fc-tunnel

To configure a Fibre Channel tunnel and facilitate RSPAN traffic, use the **interface fc-tunnel** command. To remove a configured tunnel or revert to factory defaults, use the **no** form of the command.

interface fc-tunnel {*number* **destination** *ip-address* | **explicit-path** *path-name* **source** *ip-address*}
nointerface fc-tunnel {*number* **destination** *ip-address* | **explicit-path** *path-name* **source** *ip-address*}

Syntax Description

number	Specifies a tunnel ID range from 1 to 255.
destination *ip-address*	Maps the IP address of the destination switch.
explicit-path *path-name*	Specifies a name for the explicit path. Maximum length is 16 alphanumeric characters.
source *ip-address*	Maps the IP address of the source switch.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.2(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example initiates the FC tunnel (100) in the source switch (switch S):

```
switch(config)# config terminal
switch(config)# interface fc-tunnel 100
switch(config-if)#
```

The following example maps the IP address of the source switch (switch S) to the FC tunnel (100):

```
switchS(config-if)# source 209.165.200.226
```

The following example maps the IP address of the destination switch (switch D) to the FC tunnel (100):

```
switch(config-if)# destination 209.165.200.227
```

The following example enables traffic flow through this interface:

```
switch(config-if)# no shutdown
```

The following example references the configured path in the source switch (switch S):

```
switch# config t
```

```
switch(config)# interface fc-tunnel 100
switch(config)# explicit-path Path1
```

Related Commands

Command	Description
fc-tunnel explicit-path	Configures a new or existing next-hop path.
show interface fc-tunnel	Displays an FC tunnel interface configuration for a specified interface.

interface gigabitethernet

To configure an Gigabit Ethernet interface, use the **interface gigabitethernet** command. To revert to the default values, use the **no** form of the command.

interface gigabitethernet *slot/port* **cdp enable channel-group** *group-id* [**force**] **isns** *profile-name*
nointerfacegigabitethernet *slot/port* **cdp enable channel-group isns** *profile-name*

Syntax Description

slot/port	Specifies a slot number and port number.
cdp enable	Enables Cisco Discovery Protocol (CDP) configuration parameters.
channel-group *group-id*	Adds to or removes from a PortChannel. The range is 1 to 128.
force	(Optional) Forcefully adds a port.
isns *profile-name*	Specifies the profile name to tag the interface. Maximum length is 64 characters.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(3a)	This command was introduced.
1.1(1a)	Added the **channel-group** subcommand.
1.3(1)	Added the **isns** subcommand.

Usage Guidelines

You can specify a range of interfaces by issuing a command with the following example format:

interfacegigabitethernet1/1*space-space***2***space*,*space* **gigabitethernet3/1***space-space***2**

Examples

The following example configures the Gigabit Ethernet interface at slot 4 port 1:

```
switch# config terminal
switch(config)# interface gigabitethernet 4/1
switch(config-if)#
```

The following example enters a IP address and subnet mask for the selected Gigabit Ethernet interface:

```
switch(config-if)# ip address 209.165.200.226 255.255.255.0
```

The following example changes the IP maximum transmission unit (MTU) value for the selected Gigabit Ethernet interface:

```
switch(config-if)# switchport mtu 3000
```

The following example creates a VR ID for the selected Gigabit Ethernet interface, configures the virtual IP address for the VR ID (VRRP group), and assigns a priority:

```
switch(config-if)# vrrp 100
switch(config-if-vrrp)# address 209.165.200.226
switch(config-if-vrrp)# priority 10
```

The following example adds the selected Gigabit Ethernet interface to a channel group. If the channel group does not exist, it is created, and the port is shut down:

```
switch(config-if)# channel-group 10

gigabitethernet 4/1 added to port-channel 10 and disabled
please do the same operation on the switch at the other end of the port-channel, then do
"no shutdown" at both ends to bring them up.
```

Related Commands

Command	Description
show interface	Displays an interface configuration for a specified interface.

interface ioa

To configure an IOA interface, use the **interface ioa** command. To disable this feature, use the **no** form of the command.

interface ioa {*slot*/*port*}
no interface ioa {*slot*/*port*}

Syntax Description

slot /*port*	Specifies IOA slot or port number. The range is from 1 to 16 for the slot and for the port. The range is from 1 to 4.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
NX-OS 4.2(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to configure an IOA interface for a specific cluster:

```
switch(config)# interface ioa2/1

2009 May 19 18:33:08 sjc-sw2 %IOA-2-LOG_LIBBASE_SVC_LICENSE_ON_GRACE_PERIOD: (pid=8582) No
 license. Feature will be shut down after a grace period of approximately 107 days

switch(config-if)# no shutdown
```

Related Commands

Command	Description
show ioa cluster summary	Displays the summary of all the IOA cluster.

interface iscsi

To configure an iSCSI interface, use the **interface iscsi** command. To revert to default values, use the **no** form of the command.

interface iscsi *slot/port* **mode** {**pass-thru** | **store-and-forward** | **cut-thru**} **tcp qos** *value*
nointerface iscsi *slot/port* **mode** {**pass-thru** | **store-and-forward** | **cut-thru**} **tcp qos** *value*

slot/ port	Specifies a slot number and port number.
mode	Configures a forwarding mode.
pass-thru	Forwards one frame at a time.
store-and-forward	Forwards data in one assembled unit (default).
cut-thru	Forwards one frame at a time without waiting for the exchange to complete.
tcp qos *value*	Configures the differentiated services code point (DSCP) value to apply to all outgoing IP packets. The range is 0 to 63.

Command Default

Disabled.

The TCP QoS default is 0.

The forwarding mode default is store-and-forward.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.
2.1(1)	Added the **cut-thru** option for the **mode** subcommand.

Usage Guidelines

To configure iSCSI interface, enable iSCSI using the **iscsi enable** command.

You can specify a range of interfaces by issuing a command with the following example format:

interface iscsi *space* fc1/1*space -space* 5*space ,space* fc2/5*space -space* 7

Examples

The following example enables the iSCSI feature:

```
switch# config t
switch(config)# iscsi enable
```

The following example enables the store-and-forward mode for iSCSI interfaces 9/1 to 9/4:

```
switch(config)# interface iscsi 9/1 - 4
switch(config-if)# mode store-and-forward
```

The following example reverts to using the default pass-thru mode for iSCSI interface 9/1:

```
switch(config)# interface iscsi 9/1
switch(config-if)# mode pass-thru
```

Related Commands

Command	Description
iscsi enable	Enables iSCSI.
show interface	Displays an interface configuration for a specified interface.

interface mgmt

To configure a management interface, use the **interface mgmt** command in configuration mode.

interface mgmt *number*

Syntax Description

number	Specifies the management interface number which is 0.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

When you try to shut down a management interface(mgmt0), a follow-up message confirms your action before performing the operation. Use the **force** option to bypass this confirmation, if required.

Examples

The following example configures the management interface, displays the options available for the configured interface, and exits to configuration mode:

```
switch# config terminal
switch(config)#
switch(config)# interface mgmt 0
switch(config-if)# exit
switch(config)#
```

The following example shuts down the interface without using the **force** option:

```
switch# config terminal
switch(config)# interface mgmt 0
switch(config-if)# shutdown
Shutting down this interface will drop all telnet sessions.
Do you wish to continue (y/n)? y
```

The following example shuts down the interface using the **force** option:

```
switch# config terminal
switch(config)# interface mgmt 0
switch(config-if)# shutdown force
switch(config-if)#
```

Related Commands

Command	Description
show interface mgmt	Displays interface configuration for specified interface.

interface port-channel

To configure a PortChannel interface, use the **interface port-channel** command. To remove this configuration, use the **no** form of the command.

interface port-channel *number* **channel mode active fcdomain rcf-reject vsan** *vsan-id* **fspf** [**cost** *link_cost* | **dead-interval** *seconds* | **ficon portnumber** *portnumber* | **hello-interval** *seconds* | **isns** *profile-name* | **passive** | **retransmit-interval** *seconds*]
no interface port-channel *number*

Syntax Description

number	Specifies the PortChannel number. The range is 1 to 128.
channel mode active	Configures the channel mode for the PortChannel interface.
fcdomain	Specifies the interface submode.
rcf-reject	Configures the rcf-reject flag.
vsan	Specifies the VSAN range.
vsan-id	Specifies the ID of the VSAN is from 1 to 4093.
fspf	Configures the FSPF parameters.
cost	(Optional) Configures the FSPF link cost.
link_cost	Specifies the FSPF link cost which is 1-30000.
dead-interval	(Optional) Configures the FSPF dead interval.
seconds	Specifies the dead interval (in seconds) from 2-65535.
ficon	(Optional) Configures the FICON parameters.
portnumber *portnumber*	(Optional) Configures the FICON port number for this interface.
hello-interval	(Optional) Configures FSPF hello-interval.
seconds	Specifies the hello interval (in seconds) from 1-65535.
isns	(Optional) Tags this interface to the Internet Storage Name Service (iSNS) profile.
profile-name	Specifies the profile name to tag the interface.
passive	(Optional) Enable/disable FSPF on the interface.
retransmit-interval	(Optional) Configures FSPF retransmit interface.
seconds	Specifies the retransmit interval (in seconds) from 1-65535.

Command Default

Prior to Cisco MDS NX-OS Release 8.3(1), the CLI and the Device Manager create the PortChannel in On mode in the NPIV core switches and Active mode on the NPV switches. DCNM-SAN creates all PortChannels in Active mode.

From Cisco MDS NX-OS Release 8.4(1), the CLI and the Device Manager create the PortChannel in Active mode in the NPIV core switches.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.
1.3(1)	Added **channel mode active** subcommand.
8.4(1)	This command was modified to change the default PortChannel mode from On to Active.

Usage Guidelines

Prior to Cisco MDS NX-OS Release 8.3(1), the CLI and the Device Manager create the PortChannel in On mode in the NPIV core switches and Active mode on the NPV switches. DCNM-SAN creates all PortChannels in Active mode. We recommend that you create PortChannels in Active mode.

From Cisco MDS NX-OS Release 8.4(1), the CLI and the Device Manager create the PortChannel in Active mode in the NPIV core switches.

Examples

The following example enters configuration mode and configures a PortChannel interface:

```
switch# config terminal
switch(config)# interface port-channel 32
switch(config-if)#
```

The following example assigns the FICON port number to the selected PortChannel port:

```
switch# config terminal
switch(config)# interface Port-channel 1
switch(config-if)# ficon portnumber 234
```

Related Commands

Command	Description
show interface	Displays interface configuration for specified interface.

interface sme

To configure the Cisco SME interface on a switch, use the **interface sme** command. To remove the interface, use the **no** form of the command,

interface sme *slot* / *port*
no interface sme *slot* / *port*

Syntax Description

slot	Identifies the number of the MPS-18/4 module slot.
port	Identifies the number of the Cisco SME port.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
3.2(2)	This command was introduced.

Usage Guidelines

To use this command, clustering must be enabled using the **cluster enable** command and Cisco SME services must be activated using the sme enable command.

Once you have configured the interface, use the **no shutdown** command to enable the interface.

To delete the Cisco SME interface, you must first remove the switch from the cluster. Use the **no sme cluster** command to remove the switch from the cluster and then use the **no interface** command to delete the interface.

The interface commands are available in the **(config-if)** submode.

Examples

The following example configures and enables the Cisco SME interface on the MPS-18/4 module slot and the default Cisco SME port:

```
switch# config terminal
switch(config)# interface sme 3/1
switch(config-if)# no shutdown
```

Related Commands

Command	Description
show interface sme	Displays interface information.
shutdown	Enables or disables an interface.

interface sme (Cisco SME cluster node configuration submode)

To add Cisco SME interface from a local or a remote switch to a cluster, use the **interface sme** command. To delete the interface, use the **no** form of the command.

interface sme {*slot*/*port*} [**force**]
no interface sme {*slot*/*port*} [**force**]

Syntax Description

slot	Identifies the MPS-18/4 module slot.
port	Identifies the Cisco SME port.
force	(Optional) Forcibly clears the previous interface context in the interface.

Command Default

Disabled.

Command Modes

Cisco SME cluster node configuration submode.

Command History

Release	Modification
3.2(2)	This command was introduced.

Usage Guidelines

You have to first configure a node using the **fabric-membership** command before this command can be executed.

To use this command, clustering must be enabled using the **cluster enable** command and Cisco SME services must be activated using the **sme enable** command.

To delete the Cisco SME interface, first remove the switch from the cluster. Use the **no sme cluster** command to remove the switch from the cluster and then use the **no interface** command to delete the interface.

Examples

The following example specifies the fabric to which the node belongs and then adds the Cisco SME interface (4/1) from a local switch using the force option:

```
switch# config terminal
switch(config)# sme cluster clustername1
switch(config-sme-cl)# node local
switch(config-sme-cl-node)# fabric-membership f1
switch(config-sme-cl-node)# interface sme 4/1 fabric sw-xyz
```

The following example specifies the fabric to which the node belongs and then adds the Cisco SME interface (4/1) from a remote switch using the force option:

```
switch# config terminal
switch(config)# sme cluster clustername1
switch(config-sme-cl)# node 171.71.23.33
switch(config-sme-cl-node)# fabric-membership f1
switch(config-sme-cl-node)# interface sme 4/1 fabric sw-xyz
```

Related Commands

Command	Description
fabric-membership	Adds the node to a fabric.
show interface	Displays Cisco SME interface details.

interface vsan

To configure a VSAN interface, use the **interface vsan** command. To remove a VSAN interface, use the **no** form of the command.

interface vsan *vsan-id*
no interface vsan *vsan-id*

Syntax Description

vsan-id	Specifies the VSAN ID. The range is 1 to 4093.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

None.

Examples

The following example selects a VSAN interface and enters interface configuration submode:

```
switch# config terminal
switch(config)# interface vsan 1
switch(config-if)#
```

Related Commands

Command	Description
show interface	Displays interface configuration for specified interface.

intersight connection

To configure the DNS name for intersight connection on a switch, use the **intersight connection** command. Use the **no** form of this command to not configure the DNS name for intersight connection.

intersight connection *name*
no intersight connection *name*

Syntax Description

connection	Specifies the destination name for intersight
name	Specifies the destionation host name

Command Default

Disabled

Command Modes

Configuration mode (config)

Command History

Release	Modification
9.3(2)	This command was introduced.

Usage Guidelines

None

Examples

The following example shows how to enable the Intersight feature on a switch:

```
switch# configure terminal
switch(config)# intersight connection testconnect.starshipcloud.com
```

Related Commands

Command	Description
feature intersight	Enables the feature intersight.
intersight proxy	Configures the proxy server for intersight connection.
intersight trustpoint	Configures the certificates for the intersight connection.
show system internal intersight info	Displays the device connector information.
show system internal intersight connection state	Displays the status of the connection of the devices.

intersight proxy

To configure the proxy server for the intersight connection on a switch, use the **intersight proxy** command. Use the **no** form of this command to not configure the proxy server connection.

intersight proxy *proxy-server* **port** *proxy-port*
no intersight proxy *proxy-server* **port** *proxy-port*

Syntax Description

proxy	Configure the proxy server, ipv4/ipv6/hostname.
proxy-server	IPv4 or IPv6 address or DNS name of proxy server
port	(Optional) Configure the proxy server port
proxy-port	(Optional) Proxy port number. The range is 1-65535. The default value is 8080.

Command Default

Disabled

Command Modes

Configuration mode (config)

Command History

Release	Modification
9.3(2)	This command was introduced.

Usage Guidelines

None

Examples

The following example shows how to enable the Intersight feature on a switch:

```
switch# configure terminal
switch(config)# intersight proxy proxy server proxy.esl.cisco.com port 8080
```

Related Commands

Command	Description
feature intersight	Enables the feature intersight.
intersight connection	Configures the DNS name for the intersight connection.
intersight trustpoint	Configures the certificates for the intersight connection.
show system internal intersight info	Displays the device connector information.
show system internal intersight connection state	Displays the status of the connection of the devices.

intersight trustpoint

To configure the ceritificates for intersight connection on a switch, use the **intersight trustpoint** command. Use the **no** form of this command to not configure the certificates for intersight connection.

intersight trustpoint *trustpoint-label*

no intersight trustpoint *trustpoint-label*

Syntax Description

trustpoint	Specifies the certificates for intersight
trustpoint-label	Specifies the Crypto ca truspoint label

Command Default

Disabled

Command Modes

Configuration mode (config)

Command History

Release	Modification
9.3(2)	This command was introduced.

Usage Guidelines

None

Examples

The following example shows how to enable the Intersight feature on a switch:

```
switch# configure terminal
switch(config)# intersight trustpoint mds-stage-onprem
```

Related Commands

Command	Description
feature intersight	Enables the feature intersight.
intersight proxy	Configures the proxy server for intersight connection.
intersight connection	Configures the DNS name for the intersight connection.
show system internal intersight info	Displays the device connector information.
show system internal intersight connection state	Displays the status of the connection of the devices.

ioa cluster

To configure an IOA cluster, use the **ioa cluster** command. To disable this feature, use the **no** form of the command.

ioa cluster {*cluster name*}
no ioa cluster {*cluster name*}

Syntax Description

cluster name	Specifies an IOA cluster name.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
NX-OS 4.2(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to configure an IOA cluster:

```
switch(config)# ioa cluster tape_vault
switch#(config-ioa-cl)#
```

Related Commands

Command	Description
show ioa cluster	Displays detailed information of all the IOA cluster.

ioa site-local

To configure an IOA site, use the **ioa site-local** command. To disable this feature, use the **no** form of the command.

ioa site-local {*site name*}
no ioa site-local {*site name*}

Syntax Description

site name	Specifies an IOA site name. The maximum name length is restricted to 31 alphabetical characters.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
NX-OS 4.2(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to configure an IOA local site:

```
switch# config t
switch(config)# ioa site-local SJC
switch#(config)#
```

Related Commands

Command	Description
ioa enable	Enables or disables the I/O Accelerator.

ioa-ping

To validate the connectivity between the master switch and the specified target device (for a specific flow), use the **ioa-ping** command.

ioa-ping host *hpwwn* **target** *tpwwn* **vsan** *vid* **interface** *if0*

Syntax Description

host	Specifies the host address.
hpwwn	Specifies the host PWWN for the flow.
target	Specifies the target address.
tpwwn	Specifies the target PWWN for the flow.
vsan	Specifies the VSAN.
vid	Specifies the VSAN ID. The range is from 1 to 4093.
interface	Specifies the interface associated with the flow.
if0	Specifies the ioa interface for the flow over which the test unit ready commands will be sent.

Command Default

Prompts for user input.

Command Modes

EXEC mode.

Command History

Release	Modification
NX-OS 6.2(5)	This command was introduced.

Usage Guidelines

None.

Note **ioa-ping** will work from 6.2(5) onwards and the command has to be executed from IOA master switch only.

Examples

The following example shows how to validate the connectivity between the master switch and the specified target device:

```
switch# ioa-ping host 10:00:00:00:11:a1:01:0a target 50:0a:09:80:11:4b:01:0a vsan 11 interface
 ioa 1/1

1: Round Trip Time  inf msec Device status 0
2: Round Trip Time  inf msec Device status 0
3: Round Trip Time  inf msec Device status 0
4: Round Trip Time  inf msec Device status 0
5: Round Trip Time  inf msec Device status 0
```

```
5 transmitted, 5 received ,rtt min/avg/max =  inf/ inf/ inf (msec)
switch#
```

Related Commands

Command	Description
show ioa cluster	Displays detailed information of all the IOA cluster.

ip access-group

To apply an access list to an interface, use the **ip access-group** command in interface mode. Use the **no** form of this command to negate a previously issued command or revert to factory defaults.

ip access-group *access-list-name* [**in** | **out**]

Syntax Description

access-list-name	Specifies the IP access list name. The maximum length is 64 alphanumeric characters and the text is case insensitive.
in	(Optional) Specifies that the group is for ingress traffic.
out	(Optional) Specifies that the group is for egress traffic.

Command Default

The access list is applied to both ingress and egress traffic.

Command Modes

Interface mode.

Command History

Release	Modification
1.2(1)	This command was introduced.

Usage Guidelines

The **ip access-group** command controls access to an interface. Each interface can only be associated with one access list. The access group becomes active immediately.

We recommend creating all rules in an access list, before creating the access group that uses that access list.

If you create an access group before an access list, the access list is created and all packets in that interface are dropped, because the access list is empty.

The access-group configuration for the ingress traffic applies to both local and remote traffic. The access-group configuration for the egress traffic applies only to local traffic. You can apply a different access list for each type of traffic.

Examples

The following example creates an access group called aclPermit for both the ingress and egress traffic (default):

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip access-list aclPermit permit ip any any
switch(config)# interface Gigabitethernet 3/1
switch(config-if)# ip access-group aclPermit
```

The following example deletes the access group called aclPermit:

```
switch(config-if)# no ip access-group aclPermit
```

The following example creates an access group called aclDenyTcp (if it does not already exist) for ingress traffic:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip access-list aclDenyTcp deny tcp any any
switch(config)# interface gigabitethernet 3/1
switch(config-if)# ip access-group aclDenyTcp in
```

The following example deletes the access group called aclDenyTcp for ingress traffic:

```
switch(config-if)# no ip access-group aclDenyTcp in
```

The following example creates an access list called aclPermitUdp (if it does not already exist) for local egress traffic:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip access-list aclPermitUdp permit udp 192.168.32.0 0.0.7.255 any
switch(config)# interface gigabitethernet 3/1
switch(config-if)# ip access-group aclPermitUdp out
```

The following example removes the access list called aclPermitUdp for local egress traffic:

```
switch(config-if)# no ip access-group aclPermitUdp out
```

Related Commands

Command	Description
ip access-list	Configures IP access control lists.
show ip access-list	Displays the IP-ACL configuration information.

ip access-list

IP access control lists can be used to filter IP packets though an interface. To configure IPv4 access control lists (ACLs), use the **ip access-list** command. To remove a line from an access list or completely remove the access list, use the corresponding **no** form of this command.

ip access-list *name* { **permit** | **deny** } *protocol* { **any** | *src-ip src-mask* } [*source-ports*] { **any** | *dst-ip dst-mask* } [*destination-ports*] [*options*]
no ip access-list *name* { **permit** | **deny** } *protocol* { **any** | *src-ip src-mask* } [*source-ports*] { **any** | *dst-ip dst-mask* } [*destination-ports*] [*options*]
no ip access-list *name*

where:

- *protocol*—{**icmp** | **ip** | **tcp** [**flags** {{[**ack**]} {[**all**]} {[**fin**]} {[**psh**]} {[**rst**]} {[**syn**]} {[**urg**]}}] | **udp** | *protocol-num*}
- *source-ports*—[**eq port** {**dns** | **ftp** | **ftp-data** | **http** | **ntp** | **radius** | **sftp** | **smtp** | **snmp** | **snmp-trap** | **ssh** | **syslog** | **tacacs-ds** | **tacacs-plus** | **telnet** | **tftp** | **www** | **wbem-http** | **wbem-https** | *port-num*} | **gt port** *port-num-low* | **lt port** *port-num-high* | **range port** *port-num-low port-num-high*]
- *destination-ports*—[**eq port** {**dst_dns** | **dst_ftp** | **dst_ftp-data** | **dst_http** | **dst_ntp** | **dst_radius** | **dst_sftp** | **dst_smtp** | **dst_snmp** | **dst_snmp-trap** | **dst_ssh** | **dst_syslog** | **dst_tacacs-ds** | **dst_tacacs-plus** | **dst_telnet** | **dst_tftp** | **dst_www** | **dst_wbem-http** | **dst_wbem-https** | *port-num*} | **gt port** *port-num-low* | **lt port** *port-num-high* | **range port** *port-num-low port-num-high*]
- *options*—[**established** | **icmp-type** {**echo** | **echo-reply** | **redirect** | **time-exceeded** | **unreachable** | **traceroute** | *icmp-msg-num*} [**icmp-code** *icmpcode-num*]] [**tos** {**delay** | **throughput** | **reliability** | **monetary-cost** | **normal service**}] [**log-deny**]

Syntax Description

name	Specifies an access list name. The maximum length is 28 alphanumeric characters.
deny	Drops the packet if the conditions match.
permit	Forwards the packet if the conditions match.
protocol	Specifies the name or number (integer range from 0 to 255) of an IP protocol. The IP protocol name can be **icmp**, **ip**, **tcp**, or **udp**.

flags *flag-set*	(Optional) Specifies TCP header flags to match. Multiple flags may be specified, separated by spaces. The available flag names are: **all**—Any TCP flag. **psh**—The Push flag. It indicates the data should be immediately pushed through to the receiving user. **fin**—The Finish flag. It is used to clear connections. **rst**—Reset flag. It indicates that the receiver should delete the connection without further interaction. **syn**—The Synchronize flag. It is used to establish connections. **urg**—The Urgent flag. It indicates that the urgent field is meaningful and must be added to the segment sequence number.
any	Specifies any source or destination IP address. The **any** keyword is synonymous to the address 0.0.0.0 and wildcard mask 255.255.255.255.
src-ip src-mask	Specifies the network from which the packet is sent. Mask bits are *0* for match and *1* for don't care.
dst-ip dst-mask	Specifies the network to which the packet is to be sent. Mask bits are *0* for match and *1* for don't care.

source-ports	Specifies a set of source ports to match. The syntax of this block is: *operator port-set* The following operators are available: **eq**— equal to **gt**— greater than and including **lt**— less than and including **range**— a range of source ports (inclusive) The *port-set* is a single value for the **eq**, **gt**, **lt** operators and a pair of space separated ports, in low port high port order, for the **range** operator. Ports may be specified as a number or a name. The range for numbers is 0 to 65535. The available names are as follows. TCP: **ftp-data** (20) **ftp** (21) **ssh** (22) **telnet** (23) **smtp** (25) **tacacs-plus** (49) **tacacs-ds** (65) **www** (80) **sftp** (115) **http** (143) **radius** (1812) **wbem-http** (5988) **wbem-https** (5989) UDP: **dns** (53) **tftp** (69) **ntp** (123) **snmp** (161) **snmp-trap** (162) **syslog** (514)

destination-ports	Specifies a set of destination ports to match. The syntax of this block is: *operator port-set* The following operators are available: **eq**— equal to **gt**— greater than and including **lt**— less than and including **range**— a range of source ports (inclusive) The *port-set* is a single value for the **eq**, **gt**, **lt** operators and a pair of space separated ports, in low port high port order, for the **range** operator. Ports may be specified as a number or a name. The range for numbers is 0 to 65535. The available names are as follows. TCP: **dst_ftp-data** (20) **dst_ftp** (21) **dst_ssh** (22) **dst_telnet** (23) **dst_smtp** (25) **dst_tacacs-plus** (49) **dst_tacacs-ds** (65) **dst_www** (80) **dst_sftp** (115) **dst_http** (143) **dst_radius** (1812) **dst_wbem-http** (5988) **dst_wbem-https** (5989) UDP: **dst_dns** (53) **dst_tftp** (69) **dst_ntp** (123) **dst_snmp** (161) **dst_snmp-trap** (162) **dst_syslog** (514)

icmp-type *icmp-value*	Optional) Specifies an ICMP message type to match. *icmp-value* may be a number or a name. The range for numbers is 0 to 255. The names are: **echo-reply** (0) **unreachable** (3) **redirect** (5) **echo** (8) **time-exceeded** (11) **traceroute** (30)
icmp-code *icmpcode-num*	(Optional) Specifies an ICMP message code to match as a number. The range of *icmpcode-num* is from 0 to 255.
established	(Optional) Indicates an established connection for the TCP protocol. A match occurs if the TCP datagram has the ACK, FIN, PSH, RST, or URG control bits set. The nonmatching case is that of the initial TCP datagram to form a connection.
tos *tos-value*	(Optional) Specifies the name of a type of service level to match. The names are: **normal-service** (0) **monetary-cost** (1) **reliability** (2) **throughput** (4) **delay** (8)
log-deny	(Optional) Logs an information level syslog message for each denied packet.

Command Default

No IP access lists are configured.

Command Modes

Configuration mode (config)

Command History

Release	Modification
1.2(1)	This command was introduced.

Usage Guidelines

An ACL is applied to each packet, starting at the first ACL rule. Each subsequent rule in the ACL is applied until there is a match. No further rules are applied after this. If there is no match the default rule is applied. Thus, it is important that rules are configured in the right order to achieve the desired results. Generally, 'deny' rules should be configured before 'permit' rules to ensure packets are dropped before matching an unintended 'permit' rule.

IP ACLs use an address and a wildcard mask to specify a range of IP addresses. The mask is applied to the specified address where bits in the mask that are *0* mean the corresponding bits in the specified address are

used as written (they cannot change), including *0*s. Bits that are *1* in the mask mean the corresponding bits in the address may have any value (they can change and are *wild*). This is the inverse behaviour of subnet masks.

Using the **log-deny** option at the end of the individual ACL entries shows the ACL number and whether the packet was permitted or denied, in addition to port-specific information. This option causes an information logging message about the packet that matches the dropped entry (or entries).

If the ACL specified does not exist, it is created when you enter this command. If the ACL already exists, new configuration commands are added to the end of it.

Each interface has a default action that is used when all entries in an IP ACL have been checked and there is no match. For management and non-IPS Gigabit Ethernet interfaces, this is an implicit **deny ip any any** action at the end of the IP ACL which will drop the packet. For IP Storage (IPS) interfaces, this is an implicit **permit ip any any**, which allows any IPS traffic. You must explicitly add a **deny ip any any** rule at the end of IP ACL for IPS interfaces to match the behaviour of other interfaces.

Table 2: Unsupported Keyword Combinations

Protocol Keyword	Unsupported Keywords
ip	**eq** **established** **gt** **lt** **range** **icmp-type**
icmp	**eq** **established** **gt** **lt** **range**
udp	**established** **icmp-type**
tcp	**icmp-type**

Examples

The following example configures an IP ACL called aclPermit and permits IP traffic from any source address to any destination address:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip access-list aclPermit permit ip any any
```

The following example removes the IP ACL called aclPermit:

```
switch# config terminal
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# no ip access-list aclPermit
```

The following example appends a rule to the IP ACL called aclPermit to deny TCP traffic from any source address to any destination address:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip access-list aclPermit deny tcp any any
```

The following example appends a rule to the IP ACL called aclPermitUdp that permits source addresses of 192.168.32.0 to 192.168.39.255. Subtracting 255.255.248.0 (subnet mask) from 255.255.255.255 yields 0.0.7.255:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip access-list aclPermitUdp permit udp 192.168.32.0 0.0.7.255 any
```

The following example appends a rule to the IP ACL called aclPermitIpToServer that permits all IP traffic from and to the specified networks:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip access-list aclPermitIpToServer permit ip 10.1.1.0 0.0.0.255 172.16.1.0
 0.0.0.255
```

The following example appends a rule to the IP ACL called aclDenyTcpIpPrt5 that denies TCP traffic from port 5 and any source address in the range 1.2.3.0 to 1.2.3.255 to any destination:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/
switch(config)# ip access-list aclDenyTcpIpPrt5 deny tcp 1.2.3.0 0.0.0.255 eq port 5 any
```

The following example removes this entry from the IP ACL:

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/
switch(config)# no ip access-list aclDenyTcpIpPrt5 deny tcp 1.2.3.0 0.0.0.255 eq port 5 any
```

Related Commands

Command	Description
ip access-group	Applies an IPv4 ACL to an interface.
ipv6 access-list	Configures an IPv6 access control list.
show ip access-list	Displays the configured IPv4 ACLs information.

ip address (FCIP profile configuration submode)

To assign the local IP address of a Gigabit Ethernet interface to the FCIP profile, use the **ip address** command. To remove the IP address, use the **no** form of the command.

ip address *address*
no ip address *address*

Syntax Description

address	Specifies the IP address.

Command Default

Disabled.

Command Modes

FCIP profile configuration submode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

To create a FCIP profile, you must assign a local IP address of a Gigabit Ethernet interface to the FCIP profile.

Examples

The following example assigns the local IP address of a Gigabit Ethernet interface to the FCIP profile:

```
switch# config terminal
switch(config)# fcip profile 5
switch(config-profile)# ip address 209.165.200.226
```

Related Commands

Command	Description
interface fcip interface_number use-profile profile-id	Configures the interface using an existing profile ID from 1 to 255.
show fcip profile	Displays information about the FCIP profile.

ip address (interface configuration)

To assign an IP address to a Gigabit Ethernet interface, use the **ip address** command in interface configuration submode. To remove the IP address, us the **no** form of the command.

ip address *address netmask*
no ip address *address netmask*

Syntax Description

address	Specifies the IP address.
netmask	Specifies the network mask.

Command Default

None.

Command Modes

Interface configuration submode.

Command History

Release	Modification
1.1(2)	This command was introduced.

Usage Guidelines

None.

Examples

The following example assigns an IP address to a Gigabit Ethernet interface:

```
switch# config terminal
switch(config)# interface gigabitethernet 1/2
switch(config-profile)# ip address 10.5.1.1 255.255.0.0
```

Related Commands

Command	Description
interface fcip interface_number use-profile profile-id	Configures the interface using an existing profile ID from 1 to 255.
show fcip profile	Displays information about the FCIP profile.
show interface fcip	Displays an interface configuration for a specified FCIP interface.

ip default-gateway

To configure the IP address of the default gateway, use the **ip default-gateway** command. To disable the IP address of the default gateway, use the **no** form of the command.

ip default-gateway *destination-ip-address* [**interface cpp** *slot_number/processor-number/vsan-id*]
no ip default-gateway *destination-ip-address* [**interface cpp** *slot_number/processor-number/vsan-id*]

Syntax Description

destination-ip-address	Specifies the IP address,
interface	(Optional) Configures an interface.
cpp	(Optional) Specifies a virtualization IPFC interface.
slot	(Optional) Specifies a slot number of the ASM.
processor-number	(Optional) Specifies the processor number for the IPFC interface. The current processor number is always 1.
vsan-id	(Optional) Specifies the ID of the management VSAN. The range 1 to 4093.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

None.

Examples

The following example configures the IP default gateway to 1.1.1.4:

```
switch# config terminal
switch(config)# ip default-gateway 1.1.1.4
```

Related Commands

Command	Description
show ip route	Displays the IP address of the default gateway.

ip default-network

To configure the IP address of the default network, use the **ip default-network** command in configuration mode. To disable the IP address of the default network, use the **no** form of the command.

ip default-network *ip-address*
no ip default-network *ip-address*

Syntax Description

ip-address	Specifies the IP address of the default network.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

None.

Examples

The following example configures the IP address of the default network to 1.1.1.4:

```
switch# config terminal
switch(config)# ip default-network 209.165.200.226
switch(config)# ip default-gateway 209.165.200.227
```

Related Commands

Command	Description
show ip route	Displays the IP address of the default gateway.

ip (destination-group)

To configure an IPv4 or IPv6 destination address for a destination group, use the **ip** command. To remove the destination address, use the **no** form of this command.

{**ip** | **ipv6**} **address** *address* **port** *number* [**protocol** *procedural-protocol* **encoding** *encoding-protocol*]

Syntax Description

address *address*	Destination IPv4 or IPv6 address.
port *number*	Destination port number.
protocol *procedural-protocol*	Transport protocol. gRPC is the supported transport protocol.
encoding *encoding-protocol*	Encoding format. Google Protocol Buffers (GPB) is the supported encoding format.

Command Default

IP address is not configured for a destination group.

Command Modes

Telemetry destination group configuration mode (conf-tm-dest)

Command History

Release	Modification
8.3(1)	This command was introduced.

Usage Guidelines

When the destination group is linked to a subscription node, telemetry data is sent to the IP address and port specified in the profile.

Examples

This example shows how to configure an IPv4 and IPv6 address to a destination group with the default transport protocol and default encoding:

```
switch# configure
switch(config)# telemetry
switch(config-telemetry)# destination-group 100
switch(conf-tm-dest)# ip address 1.2.3.4 port 50003 protocol gRPC encoding GPB
switch(conf-tm-dest)# destination-group 100
switch(conf-tm-dest)# ipv6 address 1:1::1:1 port 50009 protocol gRPC encoding GPB
```

This example shows how to remove an IPv4 and IPv6 address from a destination group with the default transport protocol and default encoding:

```
switch# configure
switch(config)# telemetry
switch(config-telemetry)# destination-group 100
switch(conf-tm-dest)# no ip address 1.2.3.4 port 50003 protocol gRPC encoding GPB
switch(conf-tm-dest)# destination-group 100
switch(conf-tm-dest)# no ipv6 address 1:1::1:1 port 50009 protocol gRPC encoding GPB
```

Related Commands

Command	Description
destination-group	Creates a destination group and enters destination group configuration mode.
feature telemetry	Enables the SAN Telemetry Streaming feature.
show running-config telemetry	Displays the existing telemetry configuration.
show telemetry	Displays telemetry configuration.
telemetry	Enters SAN Telemetry Streaming configuration mode.

ip domain-list

To configure or un-configure one or more domain names, use the **ip domain-list** command in configuration mode. To disable the IP domain list, use the **no** form of the command.

ip domain-list *domain-name*
no ip domain-list *domain-name*

Syntax Description

domain-name	Specifies the domain name for the IP domain list. Maximum length is 80 characters.

Command Default

If there is a domain list, the default domain name is not used.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

When “ping dino” is initiated, IP stack will append dino.cisco.com (whatever configured in domain-name) first for Name resolution. If that doesn’t succeed, it will try with domain-list.



Note If there is no domain list, the domain name that you specified with the **ip domain-name** global configuration command is used. More than one "**ip domain-list** "command can be entered and they will be tried in order.

Examples

The following example configures the IP domain list:

```
switch# config terminal
switch(config)# ip domain-list juniper.com
```

Related Commands

Command	Description
ip domain-lookup	Enables the DNS hostname to address translation.
ip name-server	Configures a list of name servers.
show ip route	Displays the IP address of the default gateway.

ip domain-lookup

To enable the DNS hostname to address translation, use the **ip domain-lookup** command in configuration mode. Use the **no** form of this command to disable this feature.

ip domain-lookup
no ip domain-lookup

Syntax Description

This command has no arguments or keywords.

Command Default

Enabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

Instead of IP addresses, you can configure the switch using meaningful names. When names are configured the switch automatically looks up the name to get its corresponding IP address.

Note In addition to **ip domain-lookup**, other commands need to be entered as well such as "**ip name-server**" and optionally, "**ip domain-name**" and "**ip domain-list**".

Examples

The following example configures a DNS server lookup feature:

```
switch# config terminal
switch(config)# ip domain-lookup
```

Related Commands

Command	Description
show ip route	Displays the IP address of the default gateway.
ip name-server	Configures a list of name servers.

ip domain-name

To configure a domain name, use the **ip domain-name** command in configuration mode. To delete a domain name, use the **no** form of the command.

ip domain-name *domain-name*
no ip domain-name *domain-name*

Syntax Description

domain-name	Specifies the domain name.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

When "**ping dino**" is initiated, IP stack will append dino.cisco.com (whatever configured in domain-name) first for name resolution. If that doesn't succeed, it will try with **domain-list**.

Examples

The following example configures a domain name:

```
switch# config terminal
switch(config)# ip domain-name cisco.com
```

Related Commands

Command	Description
ip-name server	Configures one or more IP name servers.
ip domain-list	Configure or un-configure one or more domain names.
ip domain-lookup	Enables the DNS hostname to address translation.
show ip route	Displays the IP address of the default gateway.

ip name-server

To configure one or more IP name servers, use the **ip name-server** command in configuration mode. To disable this feature, use the **no** form of the command.

ip name-server *ip-address*
no ip name-server *ip-address*

Syntax Description

ip-address	Specifies the IP address for the name server.

Command Default

The default is no name servers are configured and no IP name resolution is performed.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

You can configures a maximum of six servers. By default, no server is configured.

Examples

The following example configure a name server with an IP address of 209.165.200.226:

```
switch# config terminal
switch(config)# ip name-server 209.165.200.226
```

The following example specifies the first address (209.165.200.226) as the primary server and the second address (209.165.200.227) as the secondary sever:

```
switch(config)# ip name-server 209.165.200.226 209.165.200.227
```

The following example deletes the configured server(s) and reverts to factory default:

```
switch(config)# no ip name-server
```

Related Commands

Command	Description
ip domain-lookup	Enables the DNS hostname to address translation.
ip domain-list	Configure or un-configure one or more domain names.
ip name-server	Configures one or more IP name servers.
show ip route	Displays the IP address of the default gateway.

ip route

To configure a static route, use the **ip route** command in configuration mode.

ip route *ip-address subnet-mask* [*nexthop_ip-address*] [**interface** {**gigabitethernet** *slot /port* | **mgmt 0** | **port-channel** *channel-id* | **vsan** *vsan-id*} | **distance** *distance-number*]
no ip route *ip-address subnet-mask* [*nexthop_ip-address*] [**interface** {**gigabitethernet** *slot /port* | **mgmt 0** | **port-channel** *channel-id* | **vsan** *vsan-id*} | **distance** *distance-number*]

Syntax Description

ip-address	Specifies the IP address for the route.
subnet-mask	Specifies the subnet mask for the route.
nexthop_ip-address	(Optional) Specifies the IP address of the next hop switch.
interface	(Optional) Configures the interface associated with the route.
gigabitethernet *slot /port*	Specifies a Gigabit Ethernet interface at a port and slot.
mgmt 0	Specifies the managment interface (mgmt 0).
port-channel *channel-id*	Specifies a PortChannel interface. The range is 1 to 128.
vsan *vsan-id*	Specifies a VSAN ID. The range is 1 to 4093.
distance *distance-number*	(Optional) Specifies the distance metric for this route. It can be from 0 to 32766.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to configure a static route:

```
switch# config terminal
switch(config)# IP route 10.0.0.0 255.0.0.0 20.20.20.10 distance 10 interface vsan 1
```

Related Commands

Command	Description
show ip route	Displays the IP address routes configured in the system.

ip routing

To enable the IP forwarding feature, use the **ip routing** command in configuration mode. To disable this feature, use the **no** form of the command.

ip routing
no ip routing

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.0(2)	This command was introduced.

Usage Guidelines

None.

Examples

The following example enables the IP forwarding feature:

```
switch# config terminal
switch(config)# ip routing
```

Related Commands

Command	Description
show ip routing	Displays the IP routing state.

ip-compression

To enable compression on the FCIP link, use the **ip-compression** command in interface configuration submode. To disable compression, use the **no** form of the command.

ip-compression [**auto** | **mode1** | **mode2** | **mode3**]
no ip-compression [**auto** | **mode1** | **mode2** | **mode3**]

Syntax Description

auto	(Optional) Enables the automatic compression setting.
mode1	(Optional) Enables fast compression for the following high bandwidth links: PS-4 and IPS-8, less then 100 Mbps MPS-14/2, up to 1 Gbps
mode2	(Optional) Enables moderate compression for medium bandwidth links less than 25 Mbps.
mode3	(Optional) Enables compression for bandwidth links less than 10 Mbps.

Command Default

Disabled.

Command Modes

Interface configuration submode.

Command History

Release	Modification
1.3(1)	This command was introduced.
2.0(x)	Changed the keywords from **high-throughput** and **high-comp-ratio** to **mode1**, **mode2**, and **mode3**.

Usage Guidelines

When no compression mode is entered in the command, the default is **auto**.

The FCIP compression feature introduced in Cisco SAN-OS Release 1.3 allows IP packets to be compressed on the FCIP link if this feature is enabled on that link. By default the FCIP compression is disabled. When enabled, the software defaults to using the auto mode (if a mode is not specified).

With Cisco SAN-OS Release 2.0(1b) and later, you can configure FCIP compression using one of the following modes:

- **mode1** is a fast compression mode for high bandwidth links (> 25 Mbps).
- **mode2** is a moderate compression mode for moderately low bandwidth links (between 10 and 25 Mbps).
- **mode3** is a high compression mode for low bandwidth links (< 10 Mbps).
- **auto** (default) mode determines the appropriate compression scheme based on the bandwidth of the link (the bandwidth of the link configured in the FCIP profile's TCP parameters).

The IP compression feature behavior differs between the IPS module(s) and the MPS-14/2 module. While **mode2** and **mode3** perform software compression in both modules, **mode1** performs hardware-based compression in MPS-14/2 modules, and software compression in IPS-4 and IPS-8 modules.

In Cisco MDS SAN-OS Release 2.1(1a) and later, the **auto** mode option uses a combination of compression modes to effectively utilize the WAN bandwidth. The compression modes change dynamically to maximize the WAN bandwidth utilization.

Examples

The following example enables faster compression:

```
switch# config terminal
switch(config) interface fcip 1
switch(config-if)# ip-compression mode1
```

The following example enables automatic compression by default:

```
switch(config-if)# ip-compression
```

The following example disables compression:

```
switch(config-if)# no ip-compression
```

Related Commands

Command	Description
show interface fcip	Displays an interface configuration for a specified FCIP interface.

ips netsim delay-ms

To delay packets that arrive at a specified Gigabit Ethernet interface specifying milliseconds, use the **ips netsim delay** command in SAN extension tuner configuration submode.

ips netsim delay-ms *milliseconds* **ingress gigabitethernet** *slot/port*

Syntax Description

milliseconds	Specifies the delay in milliseconds. The range is 0 to 150.
ingress	Specifies the ingress direction.
gigabitethernet*slot/port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

This command introduces a delay for all packets entering the Gigabit Ethernet interface. Delay is unidirectional. To introduce delay in the opposite direction, use the slot and port number of the adjacent interface.

Examples

The following example shows how to configure a delay of 50 milliseconds for packets entering Gigabit Ethernet interface 2/3:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim delay-ms 50 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.
ips netsim enable	Enables the IP Network Simulator.

ips netsim delay-us

To delay packets that arrive at a specified Gigabit Ethernet interface specifying microseconds, use the **ips netsim delay** command in SAN extension tuner configuration submode.

ipsnetsimdelay-us*microseconds***ingressgigabitethernet***slot*/*port*

Syntax Description

microseconds	Specifies the delay in microseconds. The range is 0 to 150000.
ingress	Specifies the ingress direction.
gigabitethernet*slot*/*port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

This command introduces a delay for all packets entering the Gigabit Ethernet interface. Delay is unidirectional. To introduce delay in the opposite direction, use the slot and port number of the adjacent interface.

Examples

The following example shows how to configure a delay of 50 microseconds for packets entering Gigabit Ethernet interface 2/3:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim delay-us 50 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
ips netsim enable	Enables the IP Network Simulator.
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.

ips netsim drop nth

To drop packets every nth packet at a specified Gigabit Ethernet interface, use the **ips netsim drop nth** command in SAN extension tuner configuration submode.

ips netsim drop nth *packet* {**burst** *burst-size* **ingress gigabitethernet** *slot/port* | **ingress gigabitethernet** *slot/port*}

Syntax Description

packet	Specifies a specific packet to drop. The range is 0 to 10,000.
burst *burst-size*	Specifies the packet burst size. The range is 1 to 100.
ingress	Specifies the ingress direction.
gigabitethernet*slot/ port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

You can configure the IP Network Simulator to simulate packet drops (even when the queue is not full) randomly (specified as a percentage) or every Nth packet. Percentage is represented as the number of packets in 10,000. For example, if you want to drop one percent of packets, then specify it as 100 packets in 10,000. To simulate a realistic scenario for IP networks using random drops, the drop percentage should be between zero and one percent of packet drops in the specified traffic direction.

If you use the optional burst parameter, then a specified number of packets are dropped. If you do not specify the burst parameter, then only one packet is dropped. The burst limit for either random or Nth drops is 1 to 100 packets. Take the burst parameter into account when specifying the percentage of packets dropped. For example, if you select a random drop of 100 packets in 10,000 (or one percent) with a burst of 2, 200 packets (or two percent) in every 10,000 packets are dropped. Specifying 2 for burst doubles the packet drop.

Examples

The following example shows how to configure an interface to drop every 100th packet, 2 packets at a time:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim drop nth 100 burst 2 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
ips netsim enable	Enables the IP Network Simulator.
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.

ips netsim drop random

To drop packets randomly at a specified Gigabit Ethernet interface, use the **ips netsim drop random** command in SAN extension tuner configuration submode.

ips netsim drop random *packet-percentage* {**burst** *burst-size* **ingress gigabitethernet** *slot/port* | **ingress gigabitethernet** *slot/port*}

Syntax Description

packet-percentage	Specifies the percentage of packets dropped. The range is 0 to 10000.
burst *burst-size*	Specifies the packet burst size. The range is 1 to 100.
ingress	Specifies the ingress direction.
gigabitethernet*slot /port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

You can configure the IP Network Simulator to simulate packet drops (even when the queue is not full) randomly (specified as a percentage) or every Nth packet. Percentage is represented as the number of packets in 10,000. For example, if you want to drop one percent of packets, then specify it as 100 packets in 10,000. To simulate a realistic scenario for IP networks using random drops, the drop percentage should be between zero and one percent of packet drops in the specified traffic direction.

If you use the optional burst parameter, then a specified number of packets are dropped. If you do not specify the burst parameter, then only one packet is dropped. The burst limit for either random or Nth drops is 1 to 100 packets. Take the burst parameter into account when specifying the percentage of packets dropped. For example, if you select a random drop of 100 packets in 10,000 (or one percent) with a burst of 2, 200 packets (or two percent) in every 10,000 packets are dropped. Specifying 2 for burst doubles the packet drop.

Examples

The following example shows how to configure an interface to drop one percent of packets:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim drop random 100 burst 1 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.
ips netsim enable	Enables the IP Network Simulator.

ips netsim enable

To enable two Gigabit Ethernet interfaces to operate in the network simulation mode, enter the **ips netsim enable** command in SAN extension tuner configuration submode. To disable this feature, use the **no** form of the command.

ips netsim enable interface gigabitethernet *slot/port* **gigabitethernet** *slot/port*
no ips netsim enable interface gigabitethernet *slot/port* **gigabitethernet** *slot/port*

Syntax Description

interface	Specifies that interfaces are enabled.
gigabitethernet*slot/port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

This command enables two Gigabit Ethernet interfaces to simulate network characteristics. The first interface specified is the ingress port and the second interface specified is the egress port. Ports must be adjacent and the ingress interface must be an odd-numbered port.

Interfaces configured with this command can no longer be used for FCIP or iSCSI. When the SAN extension tuner configuration submode is turned off, any interface configured for network simulation reverts back to normal operation.

Examples

The following example enables the IP Network Simulator and configures interfaces 2/3 and 2/4 for network simulation:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim enable interface gigabitethernet 2/3 gigabitethernet 2/4
```

Related Commands

Command	Description
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.

ips netsim max-bandwidth-kbps

To limit the bandwidth in kilobytes per second of a specified Gigabit Ethernet interface, use the **ips netsim max-bandwidth-kbps** command in SAN extension tuner configuration submode.

ips netsim max-bandwidth-kbps *bandwidth* **ingress gigabitethernet** *slot/port*

Syntax Description

bandwidth	Specifies the bandwidth in kilobytes per second. The range is 1000 to 1000000.
ingress	Specifies the ingress direction.
gigabitethernet*slot/port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

Examples

The following example shows how to limit the interface bandwidth to 4500 Kbps:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim max-bandwidth-kbps 4500 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
ips netsim enable	Enables the IP Network Simulator.
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.

ips netsim max-bandwidth-mbps

To limit the bandwidth in megabytes per second of a specified Gigabit Ethernet interface, use the **ips netsim max-bandwidth-mbps** command in SAN extension tuner configuration submode.

ips netsim max-bandwidth-mbps *bandwidth* **ingress gigabitethernet** *slot/port*

Syntax Description

bandwidth	Specifies the bandwidth in megabytes per second. The range is 1 to 1000.
ingress	Specifies the ingress direction.
gigabitethernet*slot/port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

Examples

The following example shows how to limit the interface bandwidth to 45 Mbps:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim max-bandwidth-mbps 45 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
ips netsim enable	Enables the IP Network Simulator.
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.

ips netsim qsize

To limit the size of the queue on a specified Gigabit Ethernet interface, use the **ips netsim qsize** command in SAN extension tuner configuration submode.

ips netsim qsize *queue-size* **ingress gigabitethernet** *slot/port*

Syntax Description

queue-size	Specifies the queue size. The range is 0 to 1000000.
ingress	Specifies the ingress direction.
gigabitethernet*slot /port*	Specifies the the slot and port number of the Gigabit Ethernet interface.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

This command rate limits the size of the queue on a specified Gigabit Ethernet port. The recommended queue size for network simulation is 50000 to 150000. If the queue becomes full, packets are dropped.

Examples

The following example shows how to limit the queue size to 75 KB:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim qsize 75 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
ips netsim enable	Enables the IP Network Simulator.
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.

ips netsim reorder

To reorder packets entering a specified Gigabit Ethernet interface, use the **ips netsim reorder** command in SAN extension tuner configuration submode.

ipsnetsimreorder {**nth** *packet***distance** *dist-packet* **ingress gigabitethernet***slot*/*port*|**nth** *packet* **ingress gigabitethernet***slot*/*port*}
|{**random** *percent* **distance** *dist-packet* **ingress gigabitethernet** *slot*/*port*
|**random** *percent* **ingress gigabitethernet** *slot*/*port*}

Syntax Description

nth *packet*	Specifies a specific packet reordered. The range is 0 to 10,000.
distance *dist-packet*	Specifies the distance between the packet to be reordered and the packet at the head of the queue. The range is 1 to 10.
ingress	Specifies the ingress direction.
gigabitethernet*slot*/*port*	Specifies the the slot and port number of the Gigabit Ethernet interface.
random percent	Specifies the percentage of packets passed before a reorder. The range is 0 to 10,000.

Command Default

Disabled.

Command Modes

SAN extension tuner configuration submode.

Command History

Release	Modification
3.1(1)	This command was introduced.

Usage Guidelines

To use this command, you must enable the IP Network Simulator using the **ips netsim enable** command.

You can configure network simulator to reorder packets (even when the queue is not full) randomly (specified as a percentage) or every Nth packet. Percentage is represented as the number of packets in 10,000. For example, if you want to reorder one percent of packets, then specify it as 100 packets in 10,000. To simulate a realistic scenario for IP networks using random reordering, the percentage should be between zero and one percent of packet reordered in the specified traffic direction.

If you use the optional burst parameter, then the specified number of packets will be reordered. If you do not specify the burst parameter, then only one packet is reordered.

Examples

The following example shows reordering at 50 percent with a distance limit of 5:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
```

```
switch(config)# exit
switch#
switch# ips netsim reorder random 50 distance 5 ingress gigabitethernet 2/3
```

The following example shows reordering of every 50th packet with a distance limit of 5:

```
switch# config terminal
switch(config)#
switch(config)# san-ext-tuner enable
switch(config)# exit
switch#
switch# ips netsim reorder nth 50 distance 5 ingress gigabitethernet 2/3
```

Related Commands

Command	Description
ips netsim enable	Enables the IP Network Simulator.
show ips netsim	Displays a summary of the interfaces that are currently operating in network simulation mode.

ipv6 access-list

To configure an IPv6 access control list (ACL) and enter IPv6-ACL configuration submode, use the **ipv6 access-list** command in configuration mode. To discard an IPv6 ACL, use the **no** form of the command.

ipv6 access-list *list-name*
no ipv6 access-list *list-name*

Syntax Description

list-name	Specifies an IP access control list name. The maximum size is 64.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Before using the **ipv6 access-list** command to configure an IPv6 ACL on a switch, become familiar with the features of IPv6 and its extended addressing capabilities. In particular, it is important to understand the different types of IPv6 address formats, the IPv6 address prefix format, and the different IPv6 address types. For detailed information about IPv6.

Examples

The following example configures an IPv6 access list called List1 and enters IPv6-ACL configuration submode:

```
switch # config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ipv6 access-list List1
switch(config-ipv6-acl)#
```

The following example removes the IPv6 access list called List1 and all of its entries:

```
switch(config)# no ipv6 access-list List1
switch(config)#
```

Related Commands

ipv6 route	Configures an IPv6 static route.
ipv6 routing	Enables IPv6 unicast routing.
show ipv6 access-list	Displays a summary of ACLs.
show ipv6 route	Displays the IPv6 static routes configured on the switch.
show ipv6 routing	Displays the IPv6 unicast routing configured on the switch.

ipv6 address

To enable IPv6 processing and configure an IPv6 address on the interface, use the **ipv6 address** command in interface configuration submode. To remove an IPv6 address, use the **no** form of the command.

ipv6 address *ipv6-address-prefix*
no ipv6 address *ipv6-address-prefix*

Syntax Description

ipv6-address-prefix	Specifies the IPv6 address prefix. The format is *X:X:X::X/n* .

Command Default

None.

Command Modes

Interface configuration submode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

You can use the **ipv6 address** command to enable IPv6 processing and configure the IPv6 address on the interface. An IPv6 a ddress must be configured on an interface for the interface to forward IPv6 traffic.

Assigning a unicast address generates a link local address and implicitly enables IPv6.

Note The *ipv6-address-prefix* argument in the **ipv6 address** command must be in the form documented in R FC 2373, where the address is specified in hexadecimal using 16-bit values between colons. A slash mark (/) precedes a decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address).

Examples

The following example assigns a unicast IPv6 address to the interface and enables IPv6 processing on the interface:

```
switch#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#interface gigabitethernet 2/2
switch(config-if)#ipv6 address 2001:0DB8:800:200C::417A/64
```

Related Commands

ipv6 enable	Enables IPv6 processing on the interface.
ipv6 nd	Configures IPv6 neighbor discovery commands on the interface.
ipv6 traffic-filter	Configures IPv6 ACLs to filter traffic for packets on the interface.
show interface	Displays interface configuration information.

ipv6 enable

To enable IPv6 processing and configure an IPv6 link-local address on the interface, use the **ipv6 enable** command in interface configuration submode. To disable IPv6 processing and remove the link-local address, use the **no** form of the command.

ipv6 enable
no ipv6 enable

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

Interface configuration submode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

When you enable IPv6 on an interface, a link local address is automatically assigned. This address is used for communication on the switch:

Examples

The following example enables IPv6 processing on the interface:

```
switch#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#interface gigabitethernet 2/2
switch(config-if)#ipv6 enable
```

The following example disables IPv6 processing on the interface:

```
switch(config-if)# no ipv6 enable
```

Related Commands

ipv6 address	Configures the IPv6 address and enables IPv6 processing.
ipv6 nd	Configures IPv6 neighbor discovery commands on the interface.
ipv6 traffic-filter	Configures IPv6 ACLs to filter traffic for packets on the interface.
show interface	Displays interface configuration information.

ipv6 nd

To configure IPv6 neighbor discovery commands on the interface, use the **ipv6 nd** command in interface configuration submode. To remove IPv6 neighbor discovery configuration commands, use the **no** form of the command.

ipv6 nd {**dad attempts** *number* | **reachable-time** *time* | **retransmission-time** *time*}
no ipv6 nd {**dad attempts** *number* | **reachable-time** *time* | **retransmission-time** *time*}

Syntax Description

dad attempts *number*	Configures duplicate address detection (DAD) attempts. The range is 0 to 15.
reachable-time *time*	Configures reachability time. Specifies the reachability time in milliseconds. The range is 1000 to 3600000.
retransmission-time *time*	Configures the retransmission timer. Specifies the retransmission time in milliseconds. The range is 1000 to 3600000.

Command Default

DAD attempts: 0.

Reachable-time: 30000 milliseconds.

Retransmission-time: 1000 milliseconds.

Command Modes

Interface configuration submode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

A router must be able to determine the link-local address for each of its neighboring routers in order to ensure that the target address (the final destination) in a redirect message identifies the neighbor router by its link-local address. For static routing, the address of the next-hop router should be specified using the link-local address of the router; for dynamic routing, all IPv6 routing protocols must exchange the link-local addresses of neighboring routers.



Note A high number of DAD attempts (greater than 2) can delay address assignment.

For complete information about IPv6 neighbor discovery.

Examples

The following example sets the duplicate address detection attempts count to 2:

```
switch# config terminal
switch(config)# interface gigabitethernet 2/2
switch(config-if)# ipv6 nd dad attempts 2
```

The following example sets the reachability time to 10000 milliseconds:

```
switch(config-if)# ipv6 nd reachability-time 10000
```

The following example sets the retransmission time to 20000 milliseconds:

```
switch(config-if)# ipv6 nd retransmission-time 20000
```

Related Commands

ipv6 address	Configures the IPv6 address and enables IPv6 processing.
ipv6 enable	Enables IPv6 processing on the interface.
ipv6 traffic-filter	Configures IPv6 ACLs to filter traffic for packets on the interface.
show interface	Displays interface configuration information.

ipv6 route

To configure an IPv6 static route, use the **ipv6 route** command in configuration mode. To remove or disable an IPv6 static route, use the **no** form of the command.

ipv6 route *destination-address-prefix next-hop-address* [**distance** *distance-metric* | **interface** {**gigabitethernet** *slot/port* | **mgmt** *number* | **port-channel** *number* | **vsan** *vsan-id*}] [**distance** *distance-metric*]
no ipv6 route *destination-address-prefix next-hop-address* [**distance** *distance-metric* | **interface** {**gigabitethernet** *slot/port* | **mgmt** *number* | **port-channel** *number* | **vsan** *vsan-id*}] [**distance** *distance-metric*]

Syntax Description

destination-address-prefix	Specifies the IPv6 destination address prefix. The format is *X:X:X::X/n* .
next-hop-address	Specifies the next hop IPv6 address. The format is *X:X:X::X* .
distance	(Optional) Configures an IPv6 route metric.
distance-metric	Specifies a distance metric for the specified route. The range is 0 to 32766.
interface	(Optional) Configures a next hop IPv6 address.
gigabitethernet*slot/port*	(Optional) Specifies a Gigabit Ethernet slot and port number.
mgmt *number*	(Optional) Specifies the management interface.
port-channel *number*	(Optional) Specifies a PortChannel number. The range is 1 to 128
vsan *vsan-id*	(Optional) Specifies an IPFC VSAN ID. The range is 1 to 4093.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Before using the **ipv6 route** command to configure IPv6 features on a switch, become familiar with the features of IPv6 and its extended addressing capabilities. In particular, it is important to understand the different types of IPv6 address formats, the IPv6 address prefix format, and the different IPv6 address types.

Examples

The following example configures a static default IPv6 route on a Gigabit Ethernet interface:

```
switch # config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ipv6 route ::/0 gigabitethernet 3/1
```

The following example configures a fully specified static route on a Gigabit Ethernet interface:

```
switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet 3/2
```

The following example configures a recursive static route to a specified next hop address:

```
switch(config)# ipv6 route 2001:0DB8::/32 2001:0DB8:2002::1
```

The following example configures a recursive static route to a specified next hop address, from which the output interface is automatically derived, and to a specified interface:

```
switch(config)# ipv6 route 2001:0DB8::/32 2001:0DB8:2002::1 gigabitethernet 3/2
```

The following example configures a static IPv6 route with an administrative distance of 20.

```
switch(config)# ipv6 route 2001:0DB8::/32 interface gigabitethernet 2/0 distance 20
```

Related Commands

ipv6 access-list	Configures an IPv6 access control list (ACL) and enters IPv6-ACL configuration submode.
ipv6 routing	Enables IPv6 unicast routing.
show ipv6 access-list	Displays a summary of ACLs.
show ipv6 route	Displays the static IPv6 routes configured on the switch.
show ipv6 routing	Displays the IPv6 unicast routing configured on the switch.

ipv6 routing

To enable IPv6 unicast routing, use the **ipv6 routing** command in configuration mode. To disable IPv6 unicast routing, use the **no** form of the command.

ipv6 routing
no ipv6 routing

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Before using the **ipv6 routing** command to configure IPv6 features on a switch, become familiar with the features of IPv6 and its extended addressing capabilities. In particular, it is important to understand the different types of IPv6 address formats, the IPv6 address prefix format, and the different IPv6 address types.

Examples

The following example enables IPv6 routing:

```
switch # config terminal
switch(config)# ipv6 routing
```

The following example disables IPv6 routing:

```
switch(config)# no ipv6 routing
```

Related Commands

ipv6 access-list	Configures an IPv6 access control list (ACL) and enters IPv6-ACL configuration submode.
ipv6 route	Configures a static IPv6 route.
show ipv6 access-list	Displays a summary of ACLs.
show ipv6 route	Displays the static IPv6 routes configured on the switch.
show ipv6 routing	Displays the IPv6 unicast routing configured on the switch.

ipv6 traffic-filter

To configure IPv6 access control lists (ACLs) to filter traffic for packets on the interface, use the **ipv6 traffic-filter** command in interface configuration submode. To remove an IPv6-ACL traffic filter on the switch, use the **no** form of the command.

ipv6 traffic-filter *access-list-name* {**in** | **out**}
no ipv6 traffic-filter *access-list-name* {**in** | **out**}

Syntax Description

access-list-name	Specifies the name of an access control list for packets. The maximum size is 64 characters.
in	Configures inbound packets.
out	Configures outbound packets.

Command Default

None.

Command Modes

Interface configuration submode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example configures a traffic filter, called testfilter, for inbound packets:

```
switch# config terminal
switch(config)# interface gigabitethernet 2/2
switch(config-if)# ipv6 traffic-filter testfilter in
```

Related Commands

ipv6 address	Configures the IPv6 address and enables IPv6 processing.
ipv6 enable	Enables IPv6 processing on the interface.
ipv6 nd	Configures IPv6 ACLs to filter traffic for packets on the interface.
show interface	Displays interface configuration information.

iscsi authentication

To configure the default authentication method for iSCSI, use the **iscsi authentication** command. To revert to the default, use the **no** form of the command.

iscsi authentication {**chap** | **chap-none** | **none** | **username** *username* **password** [**0** | **7**] *password*}
no iscsi authentication {**chap** | **chap-none** | **none** | **username**}

Syntax Description

chap-none	Configures either the CHAP or no authentication.
chap	Configures the Challenge Handshake Authentication Protocol (CHAP) authentication method.
none	Specifies that no authentication is required for the selected interface
username *username*	Assigns CHAP username to be used when switch is authenticated. Specifies the name of the user. Maximum length is 128 characters.
password	Configures the password for the username.
0	(Optional) Specifies that the password is a cleartext CHAP password.
7	(Optional) Specifies that the password is an encrypted CHAP password. The password is limited to 128 characters.
password	Specifies a password for the username. The password length is limited to 128 characters.

Command Default

chap-none.

The default password is a cleartext password.

Command Modes

Configuration mode.

Command History

Release	Modification
1.1(1)	This command was introduced.
2.0(x)	Added the **username** option.

Usage Guidelines

By default, the Cisco MDS 9000 Family switch accepts an iSCSI initiator with either no authentication or CHAP authentication. If CHAP authentication is always required, use the **iscsi authentication chap** command. If no authentication is always required, use the **iscsi authentication none** command.

Use the **chap-none** option to override the global configuration which might have been configured to allow only one option either CHAP or none but not both.

Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following example configures CHAP only for ISCSI authentication:

```
switch# config terminal
switch(config)# iscsi authentication chap
```

Related Commands

Command	Description
show iscsi global	Displays all iSCSI initiators configured by the user.

iscsi duplicate-wwn-check

To check the current running configuration for conflicts between iSCSI initiators' static WWN allocation and what the system thinks is available in its WWN pool, use the **iscsi duplicate-wwn-check** command in configuration mode.

iscsi duplicate-wwn-check

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
2.1(2)	This command was introduced.

Usage Guidelines

Prior to Cisco MDS SAN-OS Release 2.1(2), WWNs assigned to static iSCSI initiators by the system can be inadvertently returned to the system when an upgrade fails or the system software is manually downgraded (that is, when you manually boot up an older Cisco MDS SAN-OS release without using the **install all** command). In these instances, the system can later assign those WWNs to other iSCSI initiators (dynamic or static) and cause conflicts.

As of Cisco MDS SAN-OS Release 2.1(2), you can use the **iscsi duplicate-wwn-check** command to check for and remove any configured WWNs that belong to the system.



Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following example shows how to check the current running configuration for conflicts between iSCSI initiators' static WWN allocation and what the system thinks is available in its WWN pool:

```
switch# config terminal
Enter configuration command, one per line. End with CNTL/Z.
switch(config)# iscsi duplicate-wwn-check

List of Potential WWN Conflicts:
-------------------------------
Node : iqn.test-local-nwnn:1-local-pwwn:1
nWWN : 22:03:00:0d:ec:02:cb:02
pWWN : 22:04:00:0d:ec:02:cb:02
```

The following example shows how to remove the conflicting nWWN and pWWN:

```
switch(config)# iscsi initiator name iqn.test-local-nwwn:1-local-pwwn:1
switch(config-iscsi-init)# no static nWWN  22:03:00:0d:ec:02:cb:02
switch(config-iscsi-init)# no static pWWN  22:04:00:0d:ec:02:cb:02
```

Related Commands

Command	Description
iscsi initiator name	Assigns an iSCSI name and changes to iSCSI initiator configuration submode.
static	Assigns persistent WWNs to an iSCSI initiator in iSCSI initiator configuration submode.
show iscsi initiator	Displays information about configured iSCSI initiators.

iscsi dynamic initiator

To configure dynamic initiator modes, use the **iscsi dynamic initiator** command in configuration mode. To revert to the default mode, use the **no** form of the command.

iscsi dynamic initiator {**deny** | **islb**}
no dynamic initiator {**deny** | **islb**}

Syntax Description

deny	Specifies that dynamic initiators are denied from logging on to the MDS switch.
islb	Specifies iSLB dynamic initiator mode.

Command Default

iSCSI.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Three dynamic initiator modes are supported:

- iSCSI—Dynamic initiators are treated as iSCSI initiators and can access dynamic virtual targets and configured iSCSI virtual targets.
- iSLB—Dynamic initiators are treated as iSLB initiators and can access dynamic virtual targets.
- Deny—Dynamic initiators are not allowed to log in to the MDS switch.

iSCSI dynamic initiator is the default mode of operation. This configuration is distributed using CFS.

Note Configuring dynamic initiator modes is supported only through the CLI, not through Device Manager or Fabric Manager.

A dynamic iSCSI initiator can be converted to a static iSCSI initiator and its WWNs can be made persistent.

A dynamic iSLB initiator can be converted to a static iSLB initiator and its WWNs can be made persistent.

Note You cannot convert a dynamic iSCSI initiator to a static iSLB initiator, or a dynamic iSLB initiator to a static iSCSI initiator.

Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following command configures the dynamic initiator mode as iSLB:

```
switch(config)# iscsi dynamic initiator islb
```

The following command configures the dynamic initiator mode as deny:

```
switch(config)# iscsi dynamic initiator deny
```

The following command reverts to the default dynamic initiator mode of iSCSI:

```
switch(config)# no iscsi dynamic initiator deny
```

Related Commands

Command	Description
iscsi save-initiator	Permanently saves the automatically assigned nWWN or pWWN mapping.
show iscsi global	Displays global iSCSI configured information.

iscsi enable

To enable the iSCSI feature in any Cisco MDS switch, use the **iscsi enable** command. To disable this feature, use the **no** form of the command.

iscsi enable
no iscsi enable

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.
3.2(2c)	Updated the example command.
NX-OS 4.1(1)	This command was deprecated.

Usage Guidelines

The configuration and verification commands for the iSCSI feature are only available when iSCSI is enabled on a switch. When you disable this feature, all related configurations are automatically discarded.

Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following command enables the iSCSI feature:

```
switch(config)# iscsi enable
switch(config)# iscsi enable module 8
switch(config)# int iscsi 2/1
switch(config-if)#
switch(config)# no shutdown
```

The following command disables the iSCSI feature (default):

```
switch(config)# no iscsi enable
```

iscsi enable module

To enable iSCSI features for each IPS linecard to create corresponding iSCSI interfaces, use the **iscsi enable module** command.

iscsi enable module *module-num*

Syntax Description

module-num	Specifies the desired IPS linecard module number on which iSCSI interfaces need to be enabled.

Command Default

iSCSI interfaces are disabled on IPS linecards by default.

Command Modes

Configuration mode.

Command History

Release	Modification
3.2(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to enable the iSCSI interface on a desired module number on the switch:

```
switch# config terminal
switch(config)# iscsi enable module 1
```

Note The iSCSI feature must be enabled before executing this command.

Related Commands

Command	Description
iscsi enable	Enables the iSCSI features but does not create the interfaces.

iscsi import target fc

To allow dynamic mapping of Fibre Channel targets, use the **iscsi import target fc** command. To disable this feature, use the **no** form of the command.

iscsi import target fc
no iscsi import target fc

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

This command directs iSCSI to dynamically import all Fibre Channel targets into iSCSI.

Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following example allows dynamic mapping of Fibre Channel targets:

```
switch# config terminal
switch(config)# iscsi import target fc
```

The following example disables dynamic mapping of Fibre Channel targets:

```
switch(config)# no iscsi import target fc
```

Related Commands

Command	Description
show iscsi global	Displays all iSCSI initiators configured by the user.

iscsi initiator idle-timeout

To configure the iSCSI initiator idle timeout, use the **iscsi initiator idle-timeout** command. To revert to the default, use the **no** form of the command.

iscsi initiator idle-timeout *seconds*
no iscsi initiator idle-timeout *seconds*

Syntax Description

seconds	Specifies the timeout in seconds. The range is 0 to 3600.

Command Default

300 seconds.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3	This command was introduced.

Usage Guidelines

When the idle timeout value is set to 0, the initiator information is cleared immediately after the last session from the initiator terminates.



Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following example configures the iSCSI initiator idle timeout to 180 seconds:

```
switch# config terminal
switch(config)# iscsi initiator idle-timeout 180
```

The following example reverts the default value of 300 seconds:

```
switch# config terminal
switch(config)# no iscsi initiator idle-timeout 240
```

Related Commands

Command	Description
show iscsi global	Displays global iSCSI configuration information.

iscsi initiator ip-address

To assign persistent WWNs to an iSCSI initiator or assign an iSCSI initiator into VSANs other than the default VSAN, use the **iscsi initiator ip-address** command. To revert to the default, use the **no** form of the command.

iscsi initiator ip-address *ipaddress* **static** {**nwwn** | **pwwn**} {*wwn-id* | **system-assign** *number*} **vsan** *vsan-id*
no iscsi initiator ip-address *ipaddress* **static** {**nwwn** | **pwwn**} {*wwn-id* | **system-assign** *number*} **vsan** *vsan-id*

Syntax Description

ipaddress	Specifies the initiator IP address.
nwwn	Configures the inititiator node WWN hex value.
pwwn	Configures the peer WWN for special frames.
wwn-id	Enters the pWWN or nWWN ID.
system-assign *number*	Generates the nWWN value automatically. The number ranges from 1 to 64.
vsan *vsan-id*	Specifies a VSAN ID. The range is 1 to 4093.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.1(1)	This command was introduced.

Usage Guidelines

Under a circumstance where an iSCSI initiator needs to have a persistent binding to FC WWNs, this command should be used. Also, an iSCSI initiator can be put into multiple VSANs. An iSCSI host can become a member of one or more VSANs.



Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following command configures an iSCSI initiator. using the IP address of the initiator node:

```
switch(config)# iscsi initiator ip address 209.165.200.226
```

The following command deletes the configured iSCSI initiator.

```
switch(config)# no iscsi initiator ip address 209.165.200.226
```

The following command uses the switch's WWN pool to allocate the nWWN for this iSCSI initiator and keeps it persistent:

```
switch(config-(iscsi-init))# static nWWN system-assign
```

The following command assigns the user provided WWN as nWWN for the iSCSI initiator. You can only specify one nWWN for each iSCSI node:

```
switch(config-(iscsi-init))# nWWN 20:00:00:05:30:00:59:11
```

The following command uses the switch's WWN pool to allocate two pWWNs for this iSCSI initiator and keeps it persistent:

```
switch(config-(iscsi-init))# static pWWN system-assign 2
```

The following command assigns the user provided WWN as pWWN for the iSCSI initiator:

```
switch(config-(iscsi-init))# pWWN 21:00:00:20:37:73:3b:20
```

Related Commands

Command	Description
show iscsi initiator	Displays information about configured iSCSI initiators.

iscsi initiator name

To configure an iSCSI initiator name and change to iSCSI configuration mode, use the **iscsi initiator name** command. To revert to factory defaults, use the **no** form of the command.

iscsi initiator name *name*
no iscsi initiator name *name*

Syntax Description

name	Enters the initiator name to be used. The minimum length is 16 characters and maximum is 223 characters.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(2)	This command was introduced.

Usage Guidelines

Under a circumstance where an iSCSI initiator needs to have a persistent binding to FC WWNs, this command should be used. Also, an iSCSI initiator can be put into multiple VSANs. An iSCSI host can become a member of one or more VSANs.

Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following example configures an iSCSI initiator using the iSCSI name of the initiator node:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# iscsi initiator name iqn.1987-02.com.cisco.initiator
```

Related Commands

Command	Description
show iscsi initiator	Displays information about configured iSCSI initiators.

iscsi interface vsan-membership

To configure VSAN membership for iSCSI interfaces, use the **iscsi interface vsan-membership** command. Use the **no** form of this command to disable this feature or to revert to factory defaults.

iscsi interface vsan-membership
no iscsi interface vsan-membership

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

If the **iscsi interface vsan-membership** command is disabled, you will not be able to configure iSCSI VSAN membership.

Caution Changing the VSAN membership, the forwarding mode, and the authentication of an iSCSI interface that is part of an iSLB VRRP group impacts load balancing on the interface.

Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following command enables the iSCSI interface VSAN membership:

```
switch# config terminal
switch(config)# iscsi interface vsan-membership
```

The following command disables the iSCSI interface VSAN membership (default):

```
switch(config)# no iscsi interface vsan-membership
```

Related Commands

Command	Description
show iscsi initiator	Displays information about configured iSCSI initiators.

iscsi save-initiator

To permanently save the automatically assigned nWWN and pWWN mapping, use the **iscsi save-initiator** command.

iscsi save-initiator [**ip-address** *ip-address* | **name** *name*]

Syntax Description

ip-address *ip-address*	(Optional) Specifies the initiator IP address.
name *name*	(Optional) Specifies the initiator name to be used from 1 to 255 characters. The minimum length is 16 characters.

Command Default

If initiator name or IP address is not specified, the nWWN and pWWN mapping for all initiators becomes permanent.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

After executing the **iscsi save-initiator** command, issue the **copy running-config startup-config** to save the nWWN and pWWN mapping across switch reboots.

After a dynamic iSCSI initiator has logged in, you may decide to permanently save the automatically assigned nWWN and pWWN mapping so this initiator uses the same mapping the next time it logs in.

You can convert a dynamic iSCSI initiator to static iSCSI initiator and make its WWNs persistent.



Note You cannot convert a dynamic iSCSI initiator to a static iSLB initiator or a dynamic iSLB initiator to a static iSCSI initiator.



Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following example shows how to save the nWWN and pWWN mapping for all the initiators:

```
switch(config)# iscsi save-initiator
```

The following example shows how to save the nWWN and pWWN mapping for an initiator named iqn.1987-02.com.cisco.initiator:

```
switch(config)# iscsi save-initiator name iqn.1987-02.com.cisco.initiator
```

Related Commands

Command	Description
iscsi initiator	Configures an iSCSI initiator.
show iscsi initiator	Displays information about configured iSCSI initiators.

iscsi virtual-target name

To create a static iSCSI virtual target, use the **iscsi virtual-target** command. To revert to the default values, use the **no** form of the command.

iscsi virtual-target name *name* **advertise interface** {**gigabitethernet** *slot/port* [*.subinterface*] | **port-channel** *channel-id* [*.subinterface*]} **all-initiator-permit initiator** {**initiator-name** | **ip-address** *ipaddress* [*netmask*]} **permit pwwn** *pwwn-id* [**fc-lun** *number* **iscsi-lun** *number* [**secondary-pwwn** *pwwn-id* [**sec-lun** *number*]] | **secondary-pwwn** *pwwn-id*] **revert-primary-port trespass**
no iscsi virtual-target name *name* **advertise interface** {**gigabitethernet** *slot/port* [*.subinterface*] | **port-channel** *channel-id* [*.subinterface*]} **all-initiator-permit initiator** {**initiator-name** | **ip-address** *ipaddress* [*netmask*]} **permit pwwn** *pwwn-id* [**fc-lun** *number* **iscsi-lun** *number* [**secondary-pwwn** *pwwn-id* [**sec-lun** *number*]] | **secondary-pwwn** *pwwn-id*] **revert-primary-port trespass**

Syntax Description

name	Enters the virtual target name to be used. The minimum length is 16 characters and maximum of 223 bytes.
advertise interface	Advertises the virtual target name on the specified interface.
gigabitethernet *slot/port subinterface*	Selects the Gigabit Ethernet interface or subinterface to configure.
port-channel *channel-id subinterface*	Selects the Port Channel interface or subinterface to configure.
all-initiator-permit	Enables all iSCSI initiator access to this target.
initiator	Configures specific iSCSI initiator access to this target.
initiator-name	Specifies the iSCSI initiator name to be used access a specified target. Maximum length is 255 characters.
ip-address *ip-address*	Specifies the iSCSI initiator IP address.
permit	Permits access to the specified target.
pwwn *pwwn-id*	Specifies the peer WWN ID for special frames.
secondary-pwwn *pwwn-id*	(Optional) Specifies the secondary pWWN ID.
fc-lun *number*	(Optional) Specifies the Fibre Channel Logical Unit Number (LUN).
iscsi-lun *number*	(Optional) Specifies the iSCSI virtual target number.
sec-lun *number*	(Optional) Specifies the secondary Fibre Channel LUN.
revert-primary-port trespass	Moves LUNs forcefully from one port to another.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.1(1)	This command was introduced.
1.3(1)	Added **revert-to-primary** and **trespass** subcommands.

Usage Guidelines

This command is used to configure a static iSCSI target for access by iSCSI initiators. A virtual target may contain a subset of LUs of an FC target or one whole FC target.

Do not specify the LUN if you want to map the whole Fibre Channel target to an iSCSI target. All Fibre Channel LUN targets are exposed to iSCSI.



Note The CLI interprets the LUN identifier value as a hexadecimal value whether or not the 0x prefix is included.

One iSCSI target cannot contain more than one Fibre Channel target.



Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following example creates a static virtual target and enters ISCSI target configuration submode:

```
switch# config terminal
switch(config)# iscsi virtual-target name 0123456789ABDEFGHI
switch(config-iscsi-tgt)#
```

The following command advertises the virtual target only on the specified interface. By default, it is advertised on all interfaces in all IPS modules.

```
switch(config-iscsi-tgt)# advertise interface gigabitethernet 4/1
```

The following command maps a virtual target node to a Fibre Channel target:

```
switch(config-iscsi-tgt)# pWWN 26:00:01:02:03:04:05:06
```

The following command enters the secondary pWWN for the virtual target node:

```
switch(config-iscsi-tgt)# pWWN 26:00:01:02:03:04:05:06 secondary-pwwn 66:00:01:02:03:04:05:02
```

Use the LUN option to map different Fibre Channel LUNs to different iSCSI virtual targets. If you have already mapped the whole Fibre Channel target, you will not be able to use this option.

```
switch(config-iscsi-tgt)# pWWN 26:00:01:02:03:04:05:06 fc-lun 0 iscsi-lun 0
```

The following command allows the specified iSCSI initiator node to access this virtual target. You can issue this command multiple times to allow multiple initiators.

```
switch(config-iscsi-tgt)# initiator iqn.1987-02.com.cisco.initiator1 permit
```

The following command prevents the specified initiator node from accessing virtual targets:

```
switch(config-iscsi-tgt)# no initiator iqn.1987-02.com.cisco.initiator1 permit
```

The following command allows the specified IP address to access this virtual target:

```
switch(config-iscsi-tgt)# initiator ip-address 209.165.200.226 permit
```

The following command prevents the specified IP address from accessing virtual targets:

```
switch(config-iscsi-tgt)# no initiator ip-address 209.165.200.226 permit
```

The following command allows all initiators in this subnetwork to access this virtual target:

```
switch(config-iscsi-tgt)# initiator ip-address 10.50.0.0 255.255.255.0 permit
```

The following command prevents all initiators in this subnetwork from accessing virtual targets:

```
switch(config-iscsi-tgt)# no initiator ip-address 10.50.0.0 255.255.255.0 permit
```

The following command allows all initiator nodes to access this virtual target:

```
switch(config-iscsi-tgt)# all-initiator-permit
```

The following command prevents any initiator node from accessing virtual targets:

```
switch(config-iscsi-tgt)# no all-initiator-permit
```

The following command configures a primary and secondary port and moves the LUNs from one port to the other using the **trespass** command:

```
switch# config terminal
switch(config)# iscsi virtual-target name iqn.1987-02.com.cisco.initiator
switch(config-iscsi-tgt)# pwwn 50:00:00:a1:94:cc secondary-pwwn 50:00:00:a1:97:ac
switch(config-iscsi-tgt)# trespass
```

Related Commands

Command	Description
show iscsi virtual target	Displays information about iSCSI virtual targets.

islb abort

To discard a pending iSCSI Server Load Balancing (iSLB) configuration, use the **islb abort** command.

islb abort

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

You can use the **islb abort** command to discard the pending changes to the iSLB configuration and release the fabric lock. This action has no effect on the active configuration on any switch in the fabric.

The **islb abort** command can be issued only by the user who started the Cisco Fabric Services (CFS) session and only on the switch that started the CFS session.

Examples

The following example discards the pending iSLB configuration distribution:

```
switch# config t
switch(config)# islb abort
```

Related Commands

Command	Description
clear islb session	Clears a pending iSLB configuration. This command can be issued on any switch by a user with admin privileges.
islb commit	Commits the iSLB configuration distribution and releases the fabric lock.
show islb cfs-session status	Displays iSLB information.
show islb pending	Displays the pending configuration changes.
show islb pending-diff	Displays the differences between the pending configuration and the current configuration.

islb commit

To commit a pending iSCSI server load balancing (iSLB) configuration, use the **islb commit** command.

islb commit

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

You can use the **islb commit** command to commit the pending changes to the iSLB configuration and release the fabric lock. This action changes the active configuration on all Cisco MDS switches in the fabric.

The **islb commit** command can be issued only by the user who started the Cisco Fabric Services (CFS) session and only on the switch that started the CFS session.

Examples

The following example commits the pending iSLB configuration distribution:

```
switch# config t
switch(config)# islb commit
```

Related Commands

Command	Description
clear islb session	Clears a pending iSLB configuration. This command can be issued on any switch by a user with admin privileges.
islb abort	Discards the pending iSLB configuration distribution and releases the fabric lock.
islb distribute	Enables iSLB configuration distribution.
show islb cfs-session status	Displays iSLB information.
show islb pending	Displays the pending configuration changes.
show islb pending-diff	Displays the differences between the pending configuration and the current configuration.

islb distribute

To enable Cisco Fabric Services for iSCSI Server Load Balancing (iSLB) configuration, use the **islb distribute** command. To disable the iSLB configuration distribution, use the **no** form of the command

islb distribute
no islb distribute

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

You can use the **islb distribute** command to enable the distribution of iSLB configuration information to other Cisco MDS switches in the fabric using the Cisco Fabric Services (CFS) infrastructure. You can synchronize the iSLB configuration across the fabric from the console of a single MDS switch.

Note The only initiator configuration that is distributed throughout the fabric using CFS is a statically mapped, iSLB initiator configuration. Dynamically mapped and statically mapped iSCSI initiator configurations are not distributed. iSCSI initiator idle-timeout and global authentication parameters are also distributed.

If you are using both iSLB and inter-VSAN routing (IVR), ensure that the following conditions are satisfied; otherwise, traffic may be disrupted in the fabric.

- You must enable both features on at least one switch in the fabric.
- You must configure and activate zoning from the switch for normal zones, IVR zones, and and iSLB zones.

Examples

The following example enables iSLB configuration distribution:

```
switch# config t
switch(config)# islb distribute
```

The following example disables iSLB configuration distribution:

```
switch(config)# no islb distribute
```

Related Commands

Command	Description
clear islb session	Clears a pending iSLB configuration. This command can be issued on any switch by a user with admin privileges.

Command	Description
islb abort	Discards the pending iSLB configuration distribution and releases the fabric lock.
islb commit	Commits the iSLB configuration distribution and releases the fabric lock.

islb initiator

To configure the iSCSI server load balancing (iSLB) initiator and enter iSLB initiator configuration submode, use the **islb initiator** command. To delete the configured ISLB initiator, use the **no** form of the command.

islb initiator {**ip-address** {*ip-addressipv6-address*} | **name** *name*}
no islb initiator name *name*

Syntax Description

ip-address	Specifies the iSLB initiator node IP address.
ip-address	Specifies the initiator IPv4 address.
ipv6-address	Specifies the initiator IPv6 address.
name *name*	Specifies the iSLB initiator node name. The maximum size is 223.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

You can use the **islb initiator** command to enter iSLB initiator configuration submode to configure static mapping for an iSLB initiator.

Examples

The following example enters iSLB initiator configuration submode to configure static mapping (using the IPv4 ip-address option) for an iSLB initiator:

```
switch# config t
switch(config)# islb initiator ipaddress 10.1.2.3
switch(config-islb-init)#
```

The following example deletes the configured iSLB initiator:

```
switch(config)# no islb initiator ipaddress 10.1.2.3
```

The following example enters iSLB initiator configuration submode to configure static mapping (using the IPv6 option) for an iSLB initiator:

```
switch# config t
switch(config)# islb initiator ipaddress 1111.2222.3333.4::5
switch(config-islb-init)#
```

The following example deletes the configured iSLB initiator:

```
switch(config)# no islb initiator ipaddress 1111.2222.3333.4::5
```

The following example enters iSLB initiator configuration submode to configure static mapping (using the name option) for an iSLB initiator:

```
switch# config t
switch(config)# islb initiator name iqn.1987-02.co..cisco.initiator
switch(config-islb-init)#
```

The following example deletes the configured iSLB initiator:

```
switch(config)# no islb initiator ipaddress name iqn.1987-02.co..cisco.initiator
```

Related Commands

Command	Description
show islb initiator configured	Displays iSLB initiator configuration information.
show islb initiator detail	Displays more detailed information about the iSLB configuration.
show islb initiator iscsi-session	Displays iSLB session details.
show islb initiator summary	Displays iSLB initiator summary information.

islb save-initiator

To permanently save the automatically assigned nWWN and pWWN mapping for the iSLB initiator, use the **islb save-initiator** command.

islb save-initiator [**ip-address** *ip-address* | **name** *name*]

Syntax Description

ip-address *ip-address*	(Optional) Specifies the initiator IP address. The format is *A.B.C.D* or *X:X:X::X* .
name *name*	(Optional) Specifies the initiator name to be used from 1 to 223 characters.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Saving the automatically assigned nWWN and pWWN mapping allows the initiator to use the same mapping the next time it logs in.

You can convert a dynamic iSLB initiator to a static iSLB initiator and make its WWNs persistent.

Note You cannot convert a dynamic iSCSI initiator to a static iSLB initiator or a dynamic iSLB initiator to a static iSCSI initiator.

Note Making the dynamic mapping for iSLB initiators static is the same as for iSCSI.

Note Only a statically mapped iSLB initiator configuration is distributed throughout the fabric using CFS. Dynamically and statically configured iSCSI initiator configurations are not distributed.

Examples

The following example saves the nWWNs and pWWNs that have automatically been assigned to the iSLB initiator whose name is specified:

```
switch# config t
switch(config)# islb save-initiator name iqn.1987-02.com.cisco.initiator
```

The following example saves the nWWNs and pWWNs that have automatically been assigned to the iSLB initiator whose IPv4 address is specified:

```
switch(config)# islb save-initiator ip-address 10.10.100.11
```

The following example saves the nWWNs and pWWNs that have automatically been assigned to all the iSLB initiators:

```
switch(config)# islb save-initiator

Please execute "copy run start" to keep the WWNs persistent across switch reboots
```

Related Commands

Command	Description
show islb session	Displays detailed iSLB session information.

islb virtual-target name

To configure an iSLB virtual target and enter iSLB target configuration submode, use the **islb virtual-target name** command. To revert to the default values, use the **no** form of the command.

islb virtual-target name *name* {**all-initiator-permit** | **initiator** {*initiator-name* **permit** | **ip address** {*A.B.C.D* **permit** | *X*:*X*:*X*::*X* **permit**}} | **pWWN permit** | **revert-primary-port permit** | **trespass permit**}
no islb virtual-target name *name* {**all-initiator-permit** | **initiator** {*initiator-name* **permit** | **ip address** {*A.B.C.D* **permit** | *X*:*X*:*X*::*X* **permit**}} | **pWWN permit** | **revert-primary-port permit** | **trespass permit**}

Syntax Description

name	Specifies the virtual target name to be used. The minimum length is 16 bytes and the maximum length is 223 bytes.
all-initiator-permit	Configures all iSLB initiators to access the target.
initiator	Configures the iSLB initiator to access the target.
initiator-name	Specifies the initiator name. The minimum length is 16 bytes and the maximum length is 223 bytes.
X:X:X::X **permit**	Permits access to the specified target.
ip address	Specifies the initiator IP address. The format is *A.B.C.D* or *X:X:X::X* .
pWWN permit	Specifies the pWWN of the Fibre Channel target. The format is *hh:hh:hh:hh:hh:hh:hh:hh* .
revert-primary-port permit	Reverts to the primary port when it becomes active again.
trespass permit	Enables trespass support.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

This command is used to configure a static target for access by iSLB initiators.

Examples

The following example creates a static virtual target and enters iSLB target configuration submode:

```
switch# config terminal
switch(config)# islb virtual-target name ABCDEFGHIJ1234567890
ips-hac1(config-islb-tgt)#
```

The following example allows all iSLB initiators to access the target:

```
ips-hac1(config-islb-tgt)# all-initiator-permit
```

The following command allows the specified IP address to access this virtual target:

```
switch(config-islb-tgt)# initiator ip-address 209.165.200.226 permit
```

The following example prevents the specified IP address from accessing virtual targets:

```
switch(config-islb-tgt)# no initiator ip-address 209.165.200.226 permit
```

The following example allows all initiators in this subnetwork to access this virtual target:

```
switch(config-islb-tgt)# initiator ip-address 10.50.0.0 255.255.255.0 permit
```

The following example prevents all initiators in this subnetwork from accessing virtual targets:

```
switch(config-islb-tgt)# no initiator ip-address 10.50.0.0 255.255.255.0 permit
```

The following example maps a pWWN to a Fibre Channel target:

```
ips-hac1(config-islb-tgt)# pwwn 26:00:01:02:03:04:05:06
```

Related Commands

Command	Description
show islb virtual-target	Displays information about iSLB virtual targets.

islb vrrp

To configure iSCSI server load balancing (iSLB) on a Virtual Router Redundancy Protocol (VRRP) group, use the **islb vrrp** command. To disable the iSLB configuration on the VRRP group, use the **no** form of the command.

islb vrrp {*group-number* **load-balance** | **ipv6** *group-number* **load-balance**}
no islb vrrp {*group-number* **load-balance** | **ipv6** *group-number* **load-balance**}

Syntax Description

group-number	Specifies an IPv4 Virtual Router group number. The range is 1 to 255.
load-balance	Enables load balancing on the VRRP group.
ipv6	Specifies IPv6 on the VRRP group.
group-number	Specifies an IPv6 Virtual Router group number. The range is 1 to 255.
load-balance	Enables load balancing on the VRRP group.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

The host is configured with a VRRP address as the portal address. When the VRRP master port receives the first iSCSI session from an initiator, it assigns a slave port to serve that particular host. The information is synchronized to all switches via Cisco Fabric Services (CFS) if recovery is needed when a master port fails. The initiator gets a temporary redirect iSCSI login response. The host then logs in to the slave port at its physical IP address. If the slave port goes down, the host will revert to the master port. The master port knows through CFS that the slave port has gone down and redirects the host to another slave port.

There are separate VRRP groups for IPv4 and IPv6. Each address family is allowed 256 virtual routers.



Note An initiator can also be redirected to the physical IP address of the master interface.



Tip The load balancing distribution is based on the number of initiators on a port and not on the number of sessions.

Caution A Gigabit Ethernet interface configured for iSLB can only be in one VRRP group because redirected sessions do not carry information about the VRRP IP address or group. This restriction allows the slave port to uniquely identify the VRRP group to which it belongs.

Caution Changing the VSAN membership, the forwarding mode, and the authentication of an iSCSI interface that is part of an iSLB VRRP group impacts load balancing on the interface.

The following example enables VRRP load balancing for IPv4 Virtual Router group 20:

```
switch# config t
switch(config)# islb vrrp 20 load-balance
```

The following example disables VRRP load balancing for IPv4 Virtual Router group 20:

```
switch(config)# no islb vrrp 20 load-balance
```

The following example enables VRRP load balancing for IPv6 Virtual Router group 30:

```
switch(config)# islb vrrp ipv6 30 load-balance
```

The following example disables VRRP load balancing for IPv6 Virtual Router group 30:

```
switch(config)# no islb ipv6 30 load-balance
```

Related Commands

Command	Description
show islb session	Displays detailed iSLB session information.

islb zoneset activate

To activate iSCSI server load balancing (iSLB) auto zones, use the **islb zoneset activate** command.

islb zoneset activate

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Auto-zoning of the initiator with the initiator targets is enabled by default.

A zone set must be active for a VSAN for auto-zones to be created in that VSAN. The **zoneset activate** command creates auto-zones only if at least one other change has been made to the zone set.

Examples

The following example activates an iSLB auto zone:

```
switch# config t
switch(config)# islb zoneset activate
```

Related Commands

Command	Description
show zoneset active	Displays active zone sets.

isns

To tag a Gigabit Ethernet or PortChannel interface to an Internet Storage Name Service (iSNS) profile, use the **isns** command in interface configuration submode. To untag the interface, use the **no** form of the command.

isns *profile-name*
no isns *profile-name*

Syntax Description

profile-name	Specifies the iSNS profile name.

Command Default

Disabled.

Command Modes

Interface configuration submode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

To use this command, iSNS must be enabled using the **isns-server enable** command.

Use the isns reregister command in EXEC mode to reregister associated iSNS objects (tagged to an iSNS profile) with the iSNS server.

Examples

The following example shows how to tag a Gigabit Ethernet interface to an iSNS profile:

```
switch# config terminal
switch(config)# interface gigabitethernet 1/2
switch(config-if)# isns Profile1
```

The following example shows how to tag a PortChannel interface to an iSNS profile:

```
switch# config terminal
switch(config)# interface port-channel 2
switch(config-if)# isns Profile2
```

Related Commands

Command	Description
isns reregister	Reregisters the iSNS object.
isns-server enable	Enables the iSNS server.
show interface gigabitethernet	Displays configuration and status information for a specified Gigabit Ethernet interface.
show interface port-channel	Displays configuration and status information for a specified PortChannel interface.
show isns	Displays iSNS information.

isns distribute

To enable Cisco Fabric Services (CFS) distribution for Internet Storage Name Service (iSNS), use the **isns distribute** command. To disable this feature, use the **no** form of the command.

isns distribute
no isns distribute

Syntax Description

This command has no other arguments or keywords.

Command Default

Enabled.

Command Modes

Configuration mode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

To use this command, iSNS must be enabled using the **isns-server enable** command.

You can configure the pWWN and nWWN of iSCSI initiators and permit a group of iSCSI initiators to share a given nWWN and pWWN pair by using a proxy initiator. The number of iSCSI initiators that register with the iSNS server is more than the number of iSCSI targets that register with the iSNS server. To synchronize the iSCSI initiator entries across switches, you can distribute the iSCSI initiator configuration to iSNS servers across switches.

Examples

The following example shows how to initiate iSNS information distribution:

```
switch# config terminal
switch(config)# isns distribute
```

The following example shows how to cancel iSNS information distribution:

```
switch# config terminal
switch(config)# no isns distribute
```

Related Commands

Command	Description
isns-server enable	Enables the iSNS server.
show isns	Displays iSNS information.

isns esi retries

To configure the number of entity status inquiry (ESI) retry attempts, use the **isns esi retries** command in configuration mode. To revert to the default value, use the **no** form of the command.

isns esi retries *number*
no isns esi retries *number*

Syntax Description

number	Specifies the number of retries. The range is 0 to 10.

Command Default

3 retries.

Command Modes

Configuration mode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

To use this command, Internet Storage Name Service (iSNS) must be enabled using the **isns-server enable** command.

The iSNS client queries the ESI port at user-configured intervals. Receipt of a response indicates that the client is still alive. Based on the configured value, the interval specifies the number of failed tries before which the client is deregistered from the server.

Examples

The following example shows how change the ESI retries limit to eight:

```
switch# config terminal
switch(config)# isns esi retries 8
```

Related Commands

Command	Description
isns-server enable	Enables the iSNS server.
show isns	Displays iSNS information.

isns profile name

To create an Internet Storage Name Service (iSNS) profile and enter iSNS profile configuration submode, use the **isns profile name** command in configuration mode. To delete the iSNS profile, use the **no** form of the command.

isns profile name *profile-name*
no isns profile name *profile-name*

Syntax Description

profile-name	Specifies the profile name. Maximum length is 64 characters.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

To use this command, iSNS must be enabled using the **isns-server enable** command.

Examples

The following example shows how to specify an iSNS profile name and enter iSNS profile configuration submode:

```
switch# config terminal
switch(config)# isns profile name UserProfile
switch(config-isns-profile)#
```

Related Commands

Command	Description
server	Configures a server IP address in an iSNS profile.
show isns	Displays iSNS information.

isns reregister

To register all Internet Storage Name Service (iSNS) objects for an interface that is already tagged to an iSNS profile, use the **isns register** command.

isns reregister {**gigabitethernet** *slot/number* | **port-channel** *channel-group*}

Syntax Description

gigabitethernet *slot/port*	Specifies tagged Gigabit Ethernet interface slot and port.
port-channel *channel-group*	Specifies tagged PortChannel group. The range is 1 to 128.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

Use this command to reregister portals and targets with the iSNS server for a tagged interface.

Examples

The following command reregisters portal and targets for a tagged interface:

```
switch# isns reregister gigabitethernet 1/4
```

Related Commands

Command	Description
show isns profile	Displays details for configured iSNS profiles.

isns-server enable

To enable the Internet Storage Name Service (iSNS) server, use the **isns-server enable** command in configuration mode. To disable iSNS, use the **no** form of the command.

isns-server enable
no isns-server enable

Syntax Description

This command has no other arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

Performing the **isns-server enable** command enables the commands used to configure iSNS.

Examples

The following example shows how to enable iSNS:

```
switch# config terminal
switch(config)# isns-server enable
```

The following example shows how to disable iSNS:

```
switch# config terminal
switch(config)# no isns-server enable
```

Related Commands

Command	Description
isns distribute	Enables iSNS distributed support.
isns esi retries	Configures ESI retry attempts.
isns profile name	Creates and configures iSNS profiles.
server	Configures iSNS server attributes.
show isns	Displays iSNS information.

ivr aam pre-deregister-check

To configure fabric precheck before deregistering IVR with AAM, use the **ivr aam pre-deregister-check** command in configuration mode.

ivr aam pre-deregister-check

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

configuration mode.

Command History

Release	Modification
NX-OS 5.0(1a)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to configure precheck before deregistering IVR with AAM:

```
switch# config terminal
switch(config)# feature ivr
switch(config-if)# ivr distribute
switch(config-if)# ivr nat
switch(config-if)# ivr commit
switch(config-if)# ivr aam pre-deregister-check
switch(config-if)#
```

Related Commands

Command	Description
show ivr aam	Displays ivr aam status.

ivr aam register

To register IVR with AAM, use the **ivr aam register** command in configuration submode. To deregister IVR with AAM, use the **no** form of the command.

ivr aam register
no ivr aam register

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

configuration mode.

Command History

Release	Modification
NX-OS 5.0(1a)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to register IVR with AAM:

```
switch# config terminal
switch(config)# feature ivr
switch(config-if)# ivr distribute
switch(config-if)# ivr nat
switch(config-if)# ivr commit
switch(config-if)# ivr aam register
switch(config-if)# 2009 Oct 20 22:12:32 isola-77 last message repeated 7 times
```

The following example shows how to deregister IVR with AAM:

```
switch# config terminal
switch(config)# feature ivr
switch(config-if)# ivr distribute
switch(config-if)# ivr nat
switch(config-if)# ivr commit
switch(config-if)# ivr aam pre-deregister-check
switch(config)# no ivr aam register

You could use "show ivr aam pre-deregister-check" to check pre-deregister status
.If the status indicates a failure, but you still go ahead with the commitment, t
he deregister might fail.
switch(config)#
```

Related Commands

Command	Description
show ivr aam	Displays IVR AAM status.

ivr abort

To discard an Inter-VSAN Routing (IVR) CFS distribution session in progress, use the **ivr abort** command in configuration mode.

ivr abort

Syntax Description

This command has no other arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to discard an IVR CFS distribution session in progress:

```
switch# config terminal
switch(config)# ivr abort
```

Related Commands

Command	Description
ivr distribute	Enables CFS distribution for IVR.
show ivr	Displays IVR CFS distribution status and other details.

ivr commit

To apply the pending configuration pertaining to the Inter-VSAN Routing (IVR) Cisco Fabric Services (CFS) distribution session in progress in the fabric, use the **ivr commit** command in configuration mode.

ivr commit

Syntax Description

This command has no other arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to apply an IVR configuration to the switches in the fabric:

```
switch# config terminal
switch(config)# ivr commit
```

Related Commands

Command	Description
ivr distribute	Enables CFS distribution for IVR.
show ivr	Displays IVR CFS distribution status and other details.

ivr copy active-service-group user-configured-service-group

To copy the active service group to the user-configured service group, use the **ivr copy active-service-group user-configured-service-group** command in EXEC mode.

ivr copy active-service-group user-configured-service-group

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example copies the active service group to the user-defined service group:

```
switch# ivr copy active-service-group user-configured-service-group
Successfully copied active service group to user-configured service group database
```

Related Commands

Command	Description
clear ivr service-group database	Clears the IVR service group database.
show ivr service-group	Displays IVR service groups.

ivr copy active-topology user-configured-topology

To copy the active inter-VSAN routing (IVR) VSAN topology to the user configured topology, use the **ivr copy active-topology user-configured-topology** command in EXEC mode.

ivr copy active-topology user-configured-topology

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

The **ivr copy active-topology user-configured-topology** command is useful if you need to edit the active IVR topology, which is not allowed. Instead you copy the active IVR topology to the user configured topology, and then edit the user configured topology.

Examples

The following example copies the active IVR topology to the user configured topology:

```
switch# ivr copy active-topology user-configured-topology

Successfully copied active VSAN-topology to user-configured topology database
```

Related Commands

Command	Description
ivr copy active-zoneset full-zoneset	Copies the active zone set to the full zone set.
ivr copy auto-topology user-configured topology	Copies the automatically discovered inter-VSAN routing (IVR) VSAN topology to the user configured topology.
show ivr vsan topology	Displays the IVR VSAN topology configuration.

ivr copy active-zoneset full-zoneset

To copy the active zone set to the full zone set, use the **ivr copy active-zoneset full-zoneset** command in EXEC mode.

ivr copy active-zoneset full-zoneset

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

Copying the active zone set to the full zone set may overwrite common zone and zone set configurations in the full zoning database.

Examples

The following example copies the active zone set to the full zone set:

```
switch# ivr copy active-zoneset full-zoneset

WARNING: This command may overwrite common zones/zonesets
         in the IVR full zoneset database
Please enter yes to proceed.(y/n) [n]?
```

Related Commands

Command	Description
ivr copy active-topology user-configured topology	Copies the active inter-VSAN routing (IVR) VSAN topology to the user configured topology.
ivr copy auto-topology user-configure topology	Copies the automatically discovered inter-VSAN routing (IVR) VSAN topology to the user configured topology.
show ivr zoneset active	Displays the active IVR zone set.

ivr copy auto-topology user-configured-topology

To copy the automatically discovered inter-VSAN routing (IVR) VSAN topology to the user configured topology, use the **ivr copy auto-topology user-configured-topology** command in EXEC mode.

ivr copy auto-topology user-configured-topology

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

EXEC configuration mode.

Command History

Release	Modification
2.1(1a)	This command was introduced.

Usage Guidelines

After using the **ivr copy auto-topology user-configured-topology** command to copy the automatically discovered VSAN topology into the user- configured topology you must use the **ivr commit** command to apply the pending configuration changes to the IVR topology using Cisco Fabric Services (CFS) distribution.

Examples

The following example copies the automatically discovered VSAN topology into the user configured topology:

```
switch# ivr copy auto-topology user-configured-topology
```

Related Commands

Command	Description
ivr commit	Applies the changes to the IVR topology.
ivr copy active-topology user-configured topology	Copies the active inter-VSAN routing (IVR) VSAN topology to the user configured topology.
ivr copy active-zoneset full-zoneset	Copies the active zone set to the full zone set.
show ivr vsan topology	Displays the IVR VSAN topology configuration .

ivr distribute

To enable Cisco Fabric Services (CFS) distribution for Inter-VSAN Routing (IVR), use the **ivr distribute** command. To disable this feature, use the **no** form of the command.

ivr distribute
no ivr distribute

Syntax Description

This command has no other arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
2.0(x)	This command was introduced.

Usage Guidelines

None.

Examples

The following example shows how to enable IVR fabric distribution:

```
switch# config terminal
switch(config)# ivr distribute
```

Related Commands

Command	Description
ivr commit	Commits temporary IVR configuration changes to the active configuration.
show ivr	Displays IVR CFS distribution status and other details.

ivr enable

To enable the Inter-VSAN Routing (IVR) feature, use the **ivr enable** command in configuration mode. To disable this feature, use the **no** form of the command.

ivr enable
no ivr enable

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.
NX-OS 4.1(1b)	This command was deprecated.

Usage Guidelines

The IVR feature must be enabled in all edge switches in the fabric that participate in the IVR.

The configuration and display commands for the IVR feature are only available when IVR is enabled on a switch.

When you disable this configuration, all related configurations are automatically discarded.

Note This command is not supported on the Cisco MDS 9124 switch, the Cisco Fabric Switch for HP c-Class BladeSystem, and the Cisco Fabric Switch for IBM BladeCenter.

Examples

The following command enters the configuration mode and enables the IVR feature on this switch:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ivr enable
```

Related Commands

Command	Description
show ivr	Displays IVR feature information.

ivr fcdomain database autonomous-fabric-num

To create IVR persistent FC IDs, use the **ivr fcdomain database autonomous-fabric-num** command. To delete the IVR fcdomain entry for a given AFID and VSAN, use the **no** form of the command.

ivr fcdomain database autonomous-fabric-num *afid-num* **vsan** *vsan-id*
no ivr fcdomain database autonomous-fabric-num *afid-num* **vsan** *vsan-id*

Syntax Description

afid-num	Specifies the current AFID. The range is 1 to 64.
vsan *vsan-id*	Specifies the current VSAN. The range is 1 to 4093.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
2.1(2)	This command was introduced.

Usage Guidelines

This configuration only takes effect when NAT mode is enabled.

Examples

The following example shows how to enter IVR fcdomain database configuration submode for AFID 10 and VSAN 20:

```
switch# config t
switch(config)# ivr fcdomain database autonomous-fabric-num 10 vsan 20
switch(config) fcdomain#
```

The following example shows how to delete all persistent FC ID database entries for AFID 10 and VSAN 20:

```
switch# config t
switch(config)# no ivr fcdomain database autonomous-fabric-num 10 vsan 20
```

Related Commands

Command	Description
show ivr fcdomain database	Displays IVR fcdomain database entry information.

ivr nat

To explicitly enable Network Address Translation (NAT) functionality for Inter-VSAN Routing (IVR), use the **ivr nat** command in configuration mode. To disable this feature, use the **no** form of the command.

ivr nat
no ivr nat

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
2.1(1a)	This command was introduced.

Usage Guidelines

The **ivr nat** command allows you to explicitly enable NAT functionality of IVR. Upgrading to SAN-OS Release 2.x from SAN-OS Release 1.3.x does not automatically enable the Fibre Channel NAT functionality. This command also allows you to continue to operate in non-NAT mode even in SAN-OS Release 2.x and later and NX-OS.



Note You might need to operate in non-NAT mode to support proprietary protocols that embed FCIDs in the frame payloads.

Examples

The following example shows how to explicitly enable NAT functionality for IVR:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ivr nat
```

Related Commands

Command	Description
show ivr	Displays IVR feature information.

ivr refresh

To refresh devices being advertised by Inter-VSAN Routing (IVR), use the **ivr refresh** command in EXEC mode.

ivr refresh

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
2.0(2)	This command was introduced.

Usage Guidelines

The **IVR refresh** command runs internally when IVR zone set or topology is activated. The limit for the maximum number of IVR zones per VSAN is 250 zones (two members per zone).

Examples

The following example shows refresh devices being advertised by IVR:

```
switch# ivr refresh
```

Related Commands

Command	Description
ivr enable	Enables the Inter-VSAN Routing (IVR) feature.
ivr withdraw domain	Withdraws an overlapping virtual domain from a specified VSAN.

ivr service-group activate

To activate an inter-VSAN routing (IVR) service group, use the **ivr service-group activate** command in configuration mode. To disable this feature, use the **no** form of the command.

ivr service-group activate [**default-sg-deny**]
no ivr service-group activate [**default-sg-deny**]

Syntax Description

default-sg-deny	(Optional) Sets the policy to deny for the default service group.

Command Default

Deactivated.

Command Modes

Configuration mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

You must activate a configured IVR service group for the IVR service group to take effect. Once a configured IVR service group is activated, it replaces the currently activated service group, if there is one.

Activating an IVR service group with the **default-sg-deny** option sets the default service group policy to deny. To change the default service group policy to allow, issue the **ivr service-group activate** command again, but without the **default-sg-deny** option.

Examples

The following example activates the default IVR service group:

```
switch# config terminal
switch(config)# ivr service-group activate
```

The following example sets the default IVR service group policy to deny:

```
switch# config terminal
switch(config)# ivr service-group activate default-sg-deny
```

The following example disables the default service group:

```
switch# config terminal
switch(config)# no ivr service-group activate
```

Related Commands

Command	Description
ivr enable	Enables inter-VSAN routing (IVR).
ivr service-group name	Configures an inter-VSAN routing (IVR) service group.
show ivr service-group database	Displays an inter-VSAN routing service group database.

ivr service-group name

To configure an Inter-VSAN Routing (IVR) service group, use the **ivr service-group name**command in configuration mode. To disable this feature, use the **no** form of the command.

ivr service-group name *service-group*
no ivr service-group name *service-group*

Syntax Description

service-group	Specifies the service group name.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
2.1(1a)	This command was introduced.

Usage Guidelines

In a complex network topology, you might only have a few IVR-enabled VSANs. To reduce the amount of traffic to non-IVR-enabled VSANs, you can configure a service group that restricts the traffic to the IVR-enabled VSANs. A service group is a combination of AFIDs and VSANs. Up to 16 service groups can be configured. A VSAN or AFID can belong to just one service group. When a new IVR-enabled switch is added to the network, you must update the service group to include the new VSANs.

There can be a maximum of 128 AFID/VSAN combinations in all service group. However, all 128 combinations can be in one service group.

The default service group ID is 0. The default service group is for all VSANs that are not a part of a user-defined service group.

Before configuring an IVR service group, you must enable the following:

- IVR using the **ivr commit** command
- IVR distribution using the **ivr commit** command
- Automatic IVR topology discovery using the **ivr commit auto command.**

Using the **autonomous-fabric-id (IVR topology database configuration)** command, you can restrict the IVR traffic to the AFIDs and VSANs configured in the service group.

Examples

The following example shows how to configure an IVR service group and change to IVR service group configuration mode:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ivr enable
switch(config)# ivr vsan-topology auto
switch(config)# ivr service-group name serviceGroup1
switch(config-ivr-sg)#
```

Related Commands

Command	Description
ivr enable	Enables the Inter-VSAN Routing (IVR) feature
ivr vsan-topology auto	Enables automatic discovery of the IVR topology.
show ivr	Displays IVR feature information.

ivr virtual-fcdomain-add

To add the Inter-VSAN Routing (IVR) virtual domains in a specific VSAN(s) to the assigned domains list in that VSAN, use the **ivr virtual-fcdomain-add** command. To delete the IVR virtual domains, use the **no** form of the command.

ivr virtual-fcdomain-add vsan-ranges *vsan-range*
no ivr virtual-fcdomain-add vsan-ranges *vsan-range*

Syntax Description

vsan-ranges *vsan-range*	Specifies the IVR VSANs or range of VSANs. The range of values for a VSAN ID is 1 to 4093.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(4)	This command was introduced.

Usage Guidelines

Use the **no ivr virtual-fcdomain-add** command to remove the currently active domains from the fcdomain manager list in a specified VSAN.

Examples

The following command adds the IVR virtual domains in VSAN:

```
switch# config terminal
switch(config)# ivr virtual-fcdomain-add vsan-ranges 1
```

The following command reverts to the factory default of not adding IVR virtual domains:

```
switch# config terminal
switch(config)# ivr virtual-fcdomain-add vsan-ranges 1
```

Related Commands

Command	Description
ivr withdraw domain	Removes overlapping domains.
show ivr virtual-fcdomain-add-status	Displays the configured VSAN topology for a fabric.

ivr virtual-fcdomain-add2

To configure the request domain_ID (RDI) mode in a specific autonomous fabric ID (AFID) and VSAN for all IVR-enabled switches, use the **ivr virtual-fcdomain-add2** command. To delete the RDI mode, use the **no** form of the command.

ivr virtual-fcdomain-add2 autonomous-fabric-id *value* **vsan-ranges** *value*
no ivr virtual-fcdomain-add2 autonomous-fabric-id *value* **vsan-ranges** *value*

Syntax Description

fabric-id *value*	Specifies the fabric ID on which the RDI mode needs to be configured.
vsan-ranges *value*	Specifies the VSAN range value on which the RDI mode needs to be configured.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
3.3(1a)	This command was introduced.

Usage Guidelines

This is a CFS distributable command.

Examples

The following example configures the RDI mode on a specific AFID and VSAN:

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch# ivr virtual-fcdomain-add2 autonomous-fabric-id 1 vsan-ranges 2
switch# fabric is now locked for configuration. Please 'commit' configuration when done.
switch(config)# ivr commit
```

Related Commands

Command	Description
show ivr virtual-fcdomain-add-status2	Displays the RDI mode in a specific AFID and VSAN for all IVR-enabled switches.

ivr vsan-topology

To configure manual or automatic discovery of the Inter-VSAN Routing (IVR) topology, use the **ivr vsan-topology** command in configuration mode.

ivr vsan-topology {**activate** | **auto**}

Syntax Description

activate	Configures manual discovery of the IVR topology and disables automatic discovery mode.
auto	Configures automatic discovery of the IVR topology.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.
2.1(1a)	Added **auto** keyword.

Usage Guidelines

To use this command you must first enable IVR using the **ivr enable** command and configure the IVR database using the **ivr vsan-topology database** command.



Caution Active IVR topologies cannot be deactivated. You can only switch to automatic topology discovery mode.

Examples

The following **ivr vsan-topology activate** command activates the VSAN topology database:

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ivr enable
switch(config)# ivr vsan-topology database
switch(config-ivr-topology-db)# autonomous-fabric-id 1 switch 20:00:00:00:30:00:3c:5e
vsan-ranges 2,2000
switch(config)# ivr vsan-topology activate
```

The following command enables VSAN topology database auto mode, which allows the switch to automatically discover the IVR topology:

```
switch(config)# ivr vsan-topology auto
```

Related Commands

Command	Description
autonomous-fabric-id(IVR topology database configuration)	Configure an autonomous phobic ID into the IVR topology database.

Command	Description
ivr enable	Enables the Inter-VSAN Routing (IVR) feature.
show ivr	Displays IVR feature information.

ivr vsan-topology auto

To configure automatic discovery of the Inter-VSAN Routing (IVR) topology, use the **ivr vsan-topology auto** command in configuration mode.

ivr vsan-topology auto

Syntax Description

This command has no arguments or keywords.

Command Default

Disabled.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

To use this command you must first enable IVR using the **ivr enable** command.

IVR configuration distribution must be enabled using the **ivr distribute** command before configuring automatic topology discovery. Once automatic IVR topology discovery is enabled, you cannot disable IVR configuration distribution.

Examples

The following command enables VSAN topology database auto mode, which allows the switch to automatically discover the IVR topology.

```
switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ivr enable
switch(config)# ivr distribute

  activate  Activate VSAN topology database for inter-VSAN routing
  auto      Enable discovery of VSAN topology for inter-VSAN routing
  database  Configure VSAN topology database for inter-VSAN routing
switch(config)# ivr vsan-topology auto
switch(config)#
```

Related Commands

Command	Description
ivr enable	Enables the Inter-VSAN Routing (IVR) feature.
autonomous-fabric-id (IVR topology database configuration)	Configure an autonomous phobic ID into the IVR topology database
show ivr	Displays IVR feature information.

ivr vsan-topology database

To configure an Inter-VSAN Routing (IVR) topology database, use the **ivr vsan-topology database** command in configuration mode. To delete an IVR topology database, use the **no** form of the command.

ivr vsan-topology database
no ivr vsan-topology database

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

To use this command you must first enable IVR using the **ivr enable** command.

You can have up to 64 VSANs (or 128 VSANs as of Cisco MDS SAN-OS Release 2.1(1a)) in an IVR topology. Specify the IVR topology using the following information:

- The switch WWNs of the IVR-enabled switches.
- A minimum of two VSANs to which the IVR-enabled switch belongs.
- The autonomous fabric ID (AFID), which distinguishes two VSANs that are logically and physically separate, but have the same VSAN number. Cisco MDS SAN-OS Release 1.3(1) and later NX-OS supports only one default AFID (AFID 1) and thus does not support non-unique VSAN IDs in the network. As of Cisco MDS SAN-OS Release 2.1(1a), you can specify up to 64 AFIDs.



Note The use of a single AFID does not allow for VSANs that are logically and physically separate but have the same VSAN number in an IVR topology.



Caution You can only configure a maximum of 128 IVR-enabled switches and 64 distinct VSANs (or 128 distinct VSANs as of Cisco MDS SAN-OS Release 2.1(1a)) in an IVR topology.

The **no ivr vsan-topology database** command only clears the configured database, not the active database. You can only delete the user-defined entries in the configured database. Auto mode entries only exist in the active database.

Examples

The following command enters configuration mode, enables the IVR feature, enters the VSAN topology database, and configures the pWWN-VSAN association for VSANs 2 and 2000:

```
switch# config terminal
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ivr enable
switch(config)# ivr vsan-topology database
switch(config-ivr-topology-db)# autonomous-fabric-id 1 switch 20:00:00:00:30:00:3c:5e
vsan-ranges 2,2000
```

Related Commands

Command	Description
autonomous0fabric-id(IVR topology database configuration)	Configures an autonomous phobic ID into the IVR topology database
ivr enable	Enables the Inter-VSAN Routing (IVR) feature.
show ivr	Displays IVR feature information.

ivr withdraw domain

To withdraw overlapping virtual domain from a specified VSAN, use the **ivr withdraw domain** command in EXEC mode.

ivr withdraw domain *domain-id* **vsan** *vsan-id*

Syntax Description

domain-id	Specifies the domain id. The range is 1 to 239.
vsan *vsan-id*	Specifies the VSAN ID. The range is 1 to 4093.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
1.3(4)	This command was introduced.

Usage Guidelines

When you enable the **ivr virtual-fcdomain-add** command, links may fail to come up due to overlapping virtual domain identifiers. If so, temporarily withdraw the overlapping virtual domain from that VSAN using the **ivr withdraw domain** command in EXEC mode.

Examples

The following command withdraws overlapping domains:

```
switch# ivr withdraw domain 10 vsan 20
```

Related Commands

Command	Description
show ivr virtual-fcdomain-add-status	Displays the configured VSAN topology for a fabric.

ivr zone name

To configure a zone for Inter-VSAN Routing (IVR), use the **ivr zone name** command. To disable a zone for IVR, use the **no** form of the command.

ivr zone name *ivzs-name*
no ivr zone name *ivz-name*

Syntax Description

ivz-name	Specifies the IVZ name. Maximum length is 59 characters.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

This command enters IVR zone configuration submode.

Examples

The following command enters the configuration mode, enables the IVR feature, creates an IVZ, and adds a pWWN-VSAN member:

```
switch# config terminal
switch(config)# ivr enable
switch(config)# ivr zone name Ivz_vsan2-3
switch(config-ivr-zone)# member pwwn 21:00:00:e0:8b:02:ca:4a vsan 3
```

Related Commands

Command	Description
show ivr	Displays IVR feature information.

ivr zone rename

To rename an inter-VSAN routing (IVR) zone, use the **ivr zone rename** command.

ivr zone rename *current-name new-name*

Syntax Description

current-name	Specifies the current zone name. The maximum size is 64 characters.
new-name	Specifies the new zone name. The maximum size is 64 characters.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example renames the IVR zone from *east* to *west:*

```
switch# ivr zone rename east west
```

Related Commands

Command	Description
ivr zone name	Creates and configures an IVR zone.
show ivr	Displays IVR information.

ivr zoneset

To configure a zoneset for Inter-VSAN Routing (IVR), use the **ivr zoneset** command. To revert to the factory defaults, use the **no** form of the command.

ivr zoneset {**activate name** *ivzs-name* [**force**] | **name** *ivzs-name*}
no ivr zoneset {**activate name** *ivzs-name* [**force**] | **name** *ivzs-name*}

Syntax Description

activate	Activates a previously configured IVZS.
force	(Optional) Forces a IVZS activation
name *ivzs-name*	Specifies the IVZS name. Maximum length is 59 characters.

Command Default

None.

Command Modes

Configuration mode.

Command History

Release	Modification
1.3(1)	This command was introduced.

Usage Guidelines

This command enters IVR zoneset configuration submode.

Note To replace the active IVR zone set with a new IVR zone set without disrupting traffic, activate the new IVR zone set without deactivating the current active IVR zone set.

Examples

The following command enters the configuration mode, enables the IVR feature, creates an IVZS, adds a IVZ member, and activates the IVZS:

```
switch# config terminal
switch(config)# ivr enable
switch(config)# ivr zoneset name Ivr_zoneset1
switch(config-ivr-zoneset)# member Ivz_vsan2-3
switch(config-ivr-zoneset)# exit
switch(config)# ivr zoneset activate name IVR_ZoneSet1
```

Related Commands

Command	Description
show ivr	Displays IVR feature information.

ivr zoneset rename

To rename an inter-VSAN routing (IVR) zone set, use the **ivr zoneset rename** command.

ivr zoneset rename *current-name new-name*

Syntax Description

current-name	Specifies the current zone set name. The maximum size is 64 characters.
new-name	Specifies the new zone set name. The maximum size is 64 characters.

Command Default

None.

Command Modes

EXEC mode.

Command History

Release	Modification
3.0(1)	This command was introduced.

Usage Guidelines

None.

Examples

The following example renames the IVR zone set from *north* to *south:*

```
switch# ivr zoneset rename north south
```

Related Commands

Command	Description
ivr zoneset name	Creates and configures an IVR zone set.
show ivr	Displays IVR information.