



## Configuring IP Storage

---

Cisco MDS 9000 Family IP storage (IPS) services modules extend the reach of Fibre Channel SANs by using open-standard, IP-based technology. The switch connects separated SAN islands using Fibre Channel over IP (FCIP), and it allows IP hosts to access Fibre Channel storage using the iSCSI protocol.

This chapter includes the following sections:

- [IP Storage Services Module, page 22-2](#)
- [Configuring Gigabit Ethernet Interfaces, page 22-4](#)
- [Configuring FCIP, page 22-19](#)
- [Configuring iSCSI, page 22-45](#)
- [iSCSI Setup Guidelines and Scenarios, page 22-85](#)
- [Configuring Storage Name Services, page 22-98](#)
- [Default Settings, page 22-101](#)



### Note

---

FCIP and iSCSI features are specific to the IPS module are available in Cisco MDS 9216 switches or Cisco MDS 9500 Directors running Cisco MDS SAN-OS Release 1.1(x) or later.

---

# IP Storage Services Module

The IP Storage (IPS) services module (IPS module) allows you to use FCIP and iSCSI features. It integrates seamlessly into the Cisco MDS 9000 Family, and it supports the full range of features available on other switching modules, including VSANs, security, and traffic management.

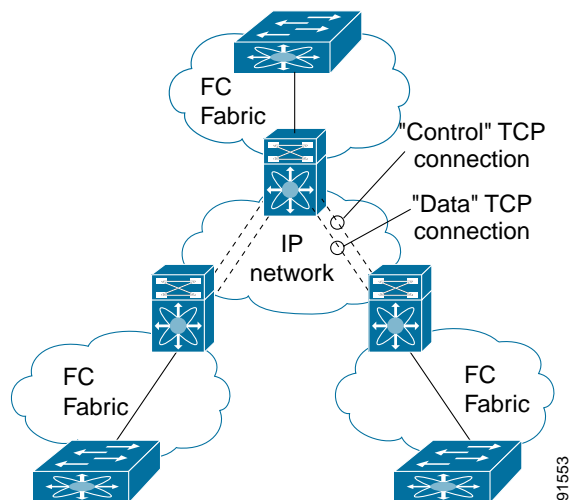
The following types of IPS modules are currently available for use in any Cisco MDS 9216 switch or any switch in the Cisco MDS 9500 Series:

- The 4-port, hot-swappable IPS module (IPS-4) has four Gigabit Ethernet ports.
- The 8-port, hot-swappable IPS module (IPS-8) has eight Gigabit Ethernet ports.

The Gigabit Ethernet ports in these module can be configured to support FCIP protocol, iSCSI protocol, or both protocols simultaneously.

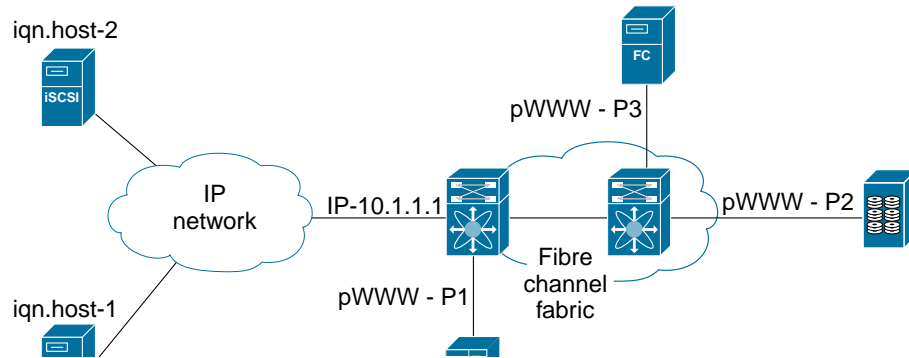
- FCIP—FCIP transports Fibre Channel frames transparently over an IP network between two Cisco MDS 9000 Family switches or other FCIP standards-compliant devices. [Figure 22-1](#) shows how the IPS module is used in different FCIP scenarios.

**Figure 22-1 FCIP Scenarios**



- iSCSI—The IPS module provides IP hosts access to Fibre Channel storage devices. The IP host sends SCSI commands encapsulated in iSCSI protocol data units (PDUs) to a Cisco MDS 9000 IPS port over a TCP/IP connection. At this point, the commands are routed from an IP network into a Fibre Channel network and forwarded to the intended target. [Figure 22-2](#) depicts the iSCSI scenarios in which the IPS module is used.

Figure 22-2 iSCSI Scenarios



## Verifying the Module Status

After inserting the module, verify the status of the module using the **show module** command:

```
switch# show module
Mod  Ports  Module-Type                Model                Status
-----
2    16     1/2 Gbps FC Module         DS-X9016             ok
3     4     IP Storage Services Module DS-X9304-SMIP        ok <-----IPS-4 module
4     8     IP Storage Services Module DS-X9308-SMIP        ok <-----IPS-8 module
5     0     Supervisor/Fabric-1        DS-X9530-SF1-K9     active *
6     0     Supervisor/Fabric-1        DS-X9530-SF1-K9     ha-standby

Mod  Sw          Hw          World-Wide-Name (s) (WWN)
-----
2    1.1(1)     0.3         20:41:00:05:30:00:86:5e to 20:50:00:05:30:00:86:5e
3    1.1(1)     0.2         20:01:00:05:30:00:86:5e to 20:c8:00:05:30:00:86:5e
4    1.1(1)     0.2         20:c1:00:05:30:00:86:5e to 20:c8:00:05:30:00:86:5e
5    1.1(1)     0.602      --
6    1.1(1)     0.602      --

Mod  MAC-Address(es)                Serial-Num
-----
2    00-05-30-00-9f-62 to 00-05-30-00-9f-66  JAB064505YV
3    00-05-30-00-ad-8e to 00-05-30-00-ad-9a JAB070806sd
4    00-05-30-00-a1-ae to 00-05-30-00-a1-ba  JAB0649059h
5    00-05-30-00-9f-f6 to 00-05-30-00-9f-fa  JAB06350B1M
6    00-05-30-00-9f-f2 to 00-05-30-00-9f-f6  JAB06350B1F
```

\* this terminal session

## Upgrading IPS Modules



### Caution

Any software upgrade for the IP Storage (IPS) services module is disruptive.

IPS modules use a rolling upgrade install mechanism whereby, each module in a given switch can only be upgraded in sequence. To guarantee a stable state, each IPS module in a switch requires a 5-minute delay before the next IPS module is upgraded.

# Configuring Gigabit Ethernet Interfaces

This section includes the following topics:

- [About Gigabit Ethernet Interfaces, page 22-4](#)
- [Configuring a Basic Gigabit Ethernet Interface, page 22-5](#)
- [About VLANs for Gigabit Ethernet, page 22-7](#)
- [Configuring the VLAN Subinterface, page 22-7](#)
- [Interface Subnet Requirements, page 22-8](#)
- [Managing IP Routing, page 22-8](#)
- [Verifying Gigabit Ethernet Connectivity, page 22-9](#)
- [Managing ARP Caches, page 22-10](#)
- [Displaying Statistics, page 22-10](#)
- [Gigabit Ethernet High Availability, page 22-14](#)
- [Configuring CDP, page 22-18](#)
- [IPS Core Dumps, page 22-18](#)

## About Gigabit Ethernet Interfaces

Both FCIP and iSCSI rely on TCP/IP for network connectivity. On each IPS module, connectivity is provided in the form of Gigabit Ethernet interfaces that are appropriately configured. This section covers the steps required to configure IP for subsequent use by FCIP and iSCSI.

A new port mode, called IPS, is defined for Gigabit Ethernet ports on each IPS module. IP storage ports are implicitly set to IPS mode, so it can only be used to perform iSCSI and FCIP storage functions. IP storage ports do not bridge Ethernet frames or route other IP packets.



**Tip**

---

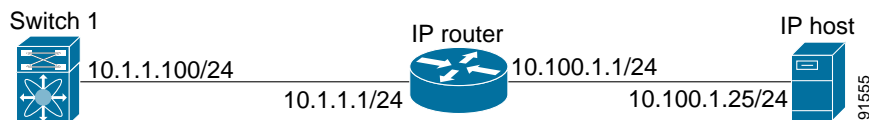
Gigabit Ethernet ports on any IPS module should not be configured in the same Ethernet broadcast domain as the management Ethernet port—they should be configured in a different broadcast domain, either by using separate standalone hubs or switches or by using separate VLANs.

---

## Configuring a Basic Gigabit Ethernet Interface

Figure 22-3 depicts a basic Gigabit Ethernet configuration.

Figure 22-3 Gigabit Ethernet Configuration



To configure the Gigabit Ethernet interface for the scenario in Figure 22-3, follow these steps:

	Command	Purpose
Step 1	switch# <b>conf t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface gigabitethernet 2/2</b> switch(config-if)#	Enters the interface configuration mode on the Gigabit Ethernet interface (slot 2, port 2).
Step 3	switch(config-if)# <b>ip address 10.1.1.100 255.255.255.0</b>	Enters the IP address (10.1.1.100) and subnet mask (255.255.255.0) for the Gigabit Ethernet interface.
Step 4	switch(config-if)# <b>no shutdown</b>	Enables the interface.

### Configuring the Interface Description

Refer to the “Configuring Interface Descriptions” section on page 10-11 for details on configuring the switchport description for any interface.

### Configuring the Beacon Mode

Refer to the “Configuring the Beacon Mode” section on page 10-14 for details on configuring the beacon mode for any interface.

### Configuring Auto-Negotiation

By default, the port is configured to auto-negotiate. You can configure auto-negotiation for a specified Gigabit Ethernet interface. By configuring auto-negotiation, the port automatically detects the speed or pause method, and duplex of incoming signals based on the link partner. You can also detect link up conditions using the auto-negotiation feature.

To configure auto-negotiation, follow these steps:

	Command	Purpose
Step 1	switch# <b>conf t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface gigabitethernet 2/2</b> switch(config-if)#	Enters the interface configuration mode on the Gigabit Ethernet interface (slot 2, port 2).

	Command	Purpose
Step 3	<code>switch(config-if)# switchport auto-negotiate</code>	Configures auto-negotiation for this Gigabit Ethernet interface. The default is off.
	<code>switch(config-if)# no switchport auto-negotiate</code>	Turns off (default) auto-negotiation for this Gigabit Ethernet interface.

## Configuring MTU Size

You can configure the switch to receive and transfer large (or jumbo) frames on a port. The default IP MTU frame size is 1500 bytes for all Ethernet ports. By configuring jumbo frames on a port, the MTU size can be increased to 9000 bytes. The following example sets the size to 3000 bytes. Independent of the MTU size, the IPS module does not pack multiple IP frames (converted to FCIP or to iSCSI).



### Note

The minimum MTU size for a port running iSCSI is 576 bytes.



### Tip

The **shutdown** or **no shutdown** command for the FCIP or iSCSI interfaces is automatically issued when you change the MTU size—you do not need to explicitly issue this command.

To configure MTU frame size, follow these steps:

	Command	Purpose
Step 1	<code>switch# config terminal</code> <code>switch(config)#</code>	Enters configuration mode.
Step 2	<code>switch(config)# interface gigabitethernet 2/2</code> <code>switch(config-if)#</code>	Enters the interface configuration mode on the Gigabit Ethernet interface (slot 2, port 2).
Step 3	<code>switch(config-if)# switchport mtu 3000</code>	Changes the IP maximum transmission unit (MTU) to 3000. The default is 1500.

## Configuring the Promiscuous Mode

You can configure the promiscuous-mode to be **on** or **off** for a specified Gigabit Ethernet interface. By configuring the promiscuous mode all packets are received by the Gigabit Ethernet interface, the software then filters and discards the packets that are not destined for that Gigabit Ethernet interface.

To configure the promiscuous mode, follow these steps:

	Command	Purpose
Step 1	<code>switch# config terminal</code> <code>switch(config)#</code>	Enters configuration mode.
Step 2	<code>switch(config)# interface gigabitethernet 2/2</code> <code>switch(config-if)#</code>	Enters the interface configuration mode on the Gigabit Ethernet interface (slot 2, port 2).

	Command	Purpose
Step 3	<code>switch(config-if)# <b>switchport promiscuous-mode on</b></code>	Configures the promiscuous mode for this Gigabit Ethernet interface. The default is off.
	<code>switch(config-if)# <b>switchport promiscuous-mode off</b></code>	Turns off (default) the promiscuous mode for this Gigabit Ethernet interface.
	<code>switch(config-if)# <b>no switchport promiscuous-mode</b></code>	Turns off (default) the promiscuous mode for this Gigabit Ethernet interface.

## About VLANs for Gigabit Ethernet

Virtual LANs (VLANs) create multiple virtual Layer 2 networks over a physical LAN network. VLANs provide traffic isolation, security, and broadcast control.

IPS Gigabit Ethernet ports automatically recognize Ethernet frames with IEEE 802.1Q VLAN encapsulation. If you need to have traffic from multiple VLANs terminated on one IPS port, configure subinterfaces—one for each VLAN.



### Note

If the IPS module is connected to a Cisco Ethernet switch, and you need to have traffic from multiple VLANs coming to one IPS port, verify the following requirements on the Ethernet switch:

- The Ethernet switch port connected to the IPS module is configured as a trunking port.
- The encapsulation is set to 802.1Q and not ISL, which is the default.

Use the VLAN ID as a subscription to the Gigabit Ethernet interface name to create the subinterface name (the slot-number>/<port-number>.<VLAN-ID>).

## Configuring the VLAN Subinterface

To configure a VLAN subinterface (the VLAN ID), follow these steps:

	Command	Purpose
Step 1	<code>switch# <b>config terminal</b></code> <code>switch(config)#</code>	Enters configuration mode.
Step 2	<code>switch(config)# <b>interface gigabitethernet 2/2.100</b></code> <code>switch(config-if)#</code>	Specifies the subinterface on which 802.1Q is used (slot 2, port 2, VLAN ID 100).  <b>Note</b> The subinterface number, 100 in this example, is the VLAN ID. The VLAN ID ranges from 1 to 4093.
Step 3	<code>switch(config-if)# <b>ip address 10.1.1.100 255.255.255.0</b></code>	Enters the IP address (10.1.1.100) and IP mask (255.255.255.0) for the Gigabit Ethernet interface.
Step 4	<code>switch(config-if)# <b>no shutdown</b></code>	Enables the interface.

## Interface Subnet Requirements

Gigabit Ethernet interfaces (major), subinterfaces (VLAN ID), and management interfaces (mgmt 0) can be configured in the same or different subnet depending on the configuration (see [Table 22-1](#)).

**Table 22-1** Subnet Requirements for Interfaces

Interface 1	Interface 2	Same Subnet Allowed	Notes
Gigabit Ethernet 1/1	Gigabit Ethernet 1/2	Yes	Two major interfaces can be configured in the same or different subnets.
Gigabit Ethernet 1/1.100	Gigabit Ethernet 1/2.100	Yes	Two subinterfaces with the same VLAN ID can be configured in the same or different subnets.
Gigabit Ethernet 1/1.100	Gigabit Ethernet 1/2.200	No	Two subinterfaces with different VLAN IDs cannot be configured in the same subnet.
Gigabit Ethernet 1/1	Gigabit Ethernet 1/1.100	No	A VLAN ID cannot be configured on the same subnet as the major interface.
mgmt0	Gigabit Ethernet 1/1.100	No	The mgmt0 interface cannot be configured in the same subnet as the Gigabit Ethernet interfaces or subinterfaces.
mgmt0	Gigabit Ethernet 1/1	No	



**Note** The configuration requirements in [Table 22-1](#) also apply to Ethernet PortChannels.

## Managing IP Routing

To configure static IP routing (see [Figure 22-3](#)) through the Gigabit Ethernet interface, follow these steps:

	Command	Purpose
Step 1	switch# <b>conf t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>ip route</b> <b>10.100.1.0 255.255.255.0 10.1.1.1</b> switch(config-if)#	Enters the IP subnet (10.100.1.0 255.255.255.0) of the IP host and configures the next hop 10.1.1.1, which is the IP address of the router connected to the Gigabit Ethernet interface.



## Displaying the IP Route Table

The **show ips ip route interface ethernet** command takes the ethernet interface as a parameter and returns the route table for the interface. See [Example 22-1](#).

### Example 22-1 Displays the Route Table

```
switch# show ips ip route interface gig 8/1
Codes: C - connected, S - static
No default gateway
C 10.1.3.0/24 is directly connected, GigabitEthernet8/1
```

Connected (C) identifies the subnet in which the interface is configured (directly connected to the interface). Static (S) identifies the static routes that go through the router.

## Verifying Gigabit Ethernet Connectivity

The **ping** command sends echo request packets out to a remote device at an IP address that you specify (see the “[Using the ping Command](#)” section on page 2-14).

Once the Gigabit Ethernet interfaces are connected with valid IP addresses, verify the interface connectivity on each switch using the **ping** command. Ping the IP host using the IP address of the host to verify that the static IP route is configured correctly. See [Example 22-2](#).

### Example 22-2 Verifying Gigabit Ethernet Connectivity

```
switch# ping 10.100.1.25
PING 10.100.1.25 (10.100.1.25): 56 data bytes
64 bytes from 10.100.1.25: icmp_seq=0 ttl=255 time=0.1 ms
64 bytes from 10.100.1.25: icmp_seq=1 ttl=255 time=0.1 ms
64 bytes from 10.100.1.25: icmp_seq=2 ttl=255 time=0.1 ms
--- 10.100.1.25 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.1 ms
```



#### Note

If the connection fails, verify the following, and repeat the **ping** command:

- The IP address for the destination (IP host) is correctly configured.
- The host is active (powered on).
- The IP route is configured correctly.
- The IP host has a route to get to the Gigabit Ethernet interface subnet.
- The Gigabit Ethernet interface is in the up state (use the **show interface gigabitethernet** command).

## Managing ARP Caches

Use the **show ips arp interface gigabitethernet** command to display the ARP cache on the Gigabit Ethernet interfaces. This command takes the Ethernet interface as a parameter and returns the ARP cache for that interface. See [Example 22-3](#).

### Example 22-3 Displays ARP Caches

```
switch# show ips arp interface gigabitethernet 7/1
Protocol      Address      Age (min)    Hardware Addr  Type   Interface
Internet     20.1.1.5     3            0005.3000.9db6 ARPA   GigabitEthernet7/1
Internet     20.1.1.10    7            0004.76eb.2ff5 ARPA   GigabitEthernet7/1
Internet     20.1.1.11    16           0003.47ad.21c4 ARPA   GigabitEthernet7/1
Internet     20.1.1.12    6            0003.4723.c4a6 ARPA   GigabitEthernet7/1
Internet     20.1.1.13    13           0004.76f0.ef81 ARPA   GigabitEthernet7/1
Internet     20.1.1.14    0            0004.76e0.2f68 ARPA   GigabitEthernet7/1
Internet     20.1.1.15    6            0003.47b2.494b ARPA   GigabitEthernet7/1
Internet     20.1.1.17    2            0003.479a.b7a3 ARPA   GigabitEthernet7/1
...
```

The ARP cache can be cleared in two ways: clearing just one entry or clearing all entries in the ARP cache. See [Examples 22-4](#) and [22-5](#).

### Example 22-4 Clearing One ARP Cache Entry

```
switch# clear ips arp address 10.2.2.2 interface gigabitethernet 8/7
arp clear successful
```

### Example 22-5 Clearing All ARP Cache Entries

```
switch# clear ips arp interface gigabitethernet 8/7
arp clear successful
```



#### Note

Use the physical interface, not the subinterface, for all ARP cache commands.

## Displaying Statistics

This section provides examples to verify Gigabit Ethernet and TCP/IP statistics on the IP storage ports.

### Displaying Gigabit Ethernet Interface Statistics

Use the **show interface Gigabit Ethernet** command on each switch to verify that the interfaces are up and functioning as desired. See [Example 22-6](#).

### Example 22-6 Displays the Gigabit Ethernet Interface

```
switch# show interface gigabitethernet 8/1
GigabitEthernet8/1 is up <-----The interface is in the up state.
  Hardware is GigabitEthernet, address is 0005.3000.a98e
  Internet address is 10.1.3.1/24
  MTU 1500 bytes, BW 1000000 Kbit
  Port mode is IPS
  Speed is 1 Gbps
```

```

Beacon is turned off
5 minutes input rate 744 bits/sec, 93 bytes/sec, 1 frames/sec
5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
3343 packets input, 406582 bytes
  0 multicast frames, 0 compressed
  0 input errors, 0 frame, 0 overrun 0 fifo
8 packets output, 336 bytes, 0 underruns
  0 output errors, 0 collisions, 0 fifo
  0 carrier errors

```

### Example 22-7 Displays the Gigabit Ethernet Subinterface

```

switch# show interface gigabitethernet 4/2.100
GigabitEthernet4/2.100 is up
  Hardware is GigabitEthernet, address is 0005.3000.abcb
  Internet address is 10.1.2.100/24
  MTU 1500 bytes
  5 minutes input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
  5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
  0 packets input, 0 bytes
    0 multicast frames, 0 compressed
    0 input errors, 0 frame, 0 overrun 0 fifo
  1 packets output, 46 bytes, 0 underruns
    0 output errors, 0 collisions, 0 fifo
    0 carrier errors

```

## Displaying Ethernet MAC Statistics

The **show ips stats mac interface gigabitethernet** command takes the main Gigabit Ethernet interface as a parameter and returns Ethernet statistics for that interface. See [Example 22-8](#).

### Example 22-8 Displays Ethernet MAC Statistics

```

switch# show ips stats mac interface gigabitethernet 8/1
Ethernet MAC statistics for port GigabitEthernet8/1
  Hardware Transmit Counters
    237 frame 43564 bytes
    0 collisions, 0 late collisions, 0 excess collisions
    0 bad frames, 0 FCS error, 0 abort, 0 runt, 0 oversize
  Hardware Receive Counters
    427916 bytes, 3464 frames, 0 multicasts, 3275 broadcasts
    0 bad, 0 runt, 0 CRC error, 0 length error
    0 code error, 0 align error, 0 oversize error
  Software Counters
    3429 received frames, 237 transmit frames
    0 frames soft queued, 0 current queue, 0 max queue
    0 dropped, 0 low memory

```



#### Note

Use the physical interface, not the subinterface, to display Ethernet MAC statistics.

## Displaying DMA-Bridge Statistics

You can display direct memory access (DMA) device statistics using the **show ips stats dma-bridge interface gigabitethernet** command. This command takes the main Gigabit Ethernet interface as a parameter and returns Ethernet statistics for that interface. See [Example 22-9](#).

### Example 22-9 Displays DMA-Bridge Statistics

```
switch# show ips stats dma-bridge interface gigabitethernet 7/1
Dma-bridge ASIC Statistics for port GigabitEthernet7/1
  Hardware Egress Counters
    231117 Good, 0 bad protocol, 0 bad header cksum, 0 bad FC CRC
  Hardware Ingress Counters
    218255 Good, 0 protocol error, 0 header checksum error
    0 FC CRC error, 0 iSCSI CRC error, 0 parity error
  Software Egress Counters
    231117 good frames, 0 bad header cksum, 0 bad FIFO SOP
    0 parity error, 0 FC CRC error, 0 timestamp expired error
    0 unregistered port index, 0 unknown internal type
    0 RDL ok, 0 RDL drop (too big), 0 RDL ttl_1
    3656368645 idle poll count, 0 loopback, 0 FCC PQ, 0 FCC EQ
    Flow Control: 0 [0], 0 [1], 0 [2], 0 [3]
  Software Ingress Counters
    218255 Good frames, 0 header cksum error, 0 FC CRC error
    0 iSCSI CRC error, 0 descriptor SOP error, 0 parity error
    0 frames soft queued, 0 current Q, 0 max Q, 0 low memory
    0 out of memory drop, 0 queue full drop
    0 RDL ok, 0 RDL drop (too big)
    Flow Control: 0 [0], 0 [1], 0 [2], 0 [3]
```

This output shows all Fibre Channel frames that ingress or egress from the Gigabit Ethernet port.



Note

---

Use the physical interface, not the subinterface, to display DMA-bridge statistics.

---

## Displaying TCP/IP Statistics



Note

---

Use the physical interface, not the subinterface, to display TCP/IP statistics.

---

Use the **show ips stats ip interface gigabitethernet** to display and verify IP statistics. This command takes the main Gigabit Ethernet interface as a parameter and returns IP statistics for that interface. See [Example 22-10](#).

### Example 22-10 Displays IP Statistics

```
switch# show ips stats ip interface gigabitethernet 4/1
Internet Protocol Statistics for port GigabitEthernet4/1
  168 total received, 168 good, 0 error
  0 reassembly required, 0 reassembled ok, 0 dropped after timeout
  371 packets sent, 0 outgoing dropped, 0 dropped no route
  0 fragments created, 0 cannot fragment
```

Use the **show ips stats tcp interface gigabitethernet** to display and verify TCP statistics. This command takes the main Ethernet interface as a parameter, and shows TCP stats along with the connection list and TCP state. The **detail** option shows all information maintained by the interface. See Examples 22-11 and 22-12.

#### Example 22-11 Displays TCP Statistics

```
switch# show ips stats tcp interface gigabitethernet 4/1
TCP Statistics for port GigabitEthernet4/1
Connection Stats
  0 active openings, 3 accepts
  0 failed attempts, 12 reset received, 3 established
Segment stats
  163 received, 355 sent, 0 retransmitted
  0 bad segments received, 0 reset sent
TCP Active Connections
  Local Address      Remote Address      State      Send-Q  Recv-Q
  0.0.0.0:3260      0.0.0.0:0          LISTEN    0       0
```

#### Example 22-12 Displays Detailed TCP Statistics

```
switch# show ips stats tcp interface gigabitethernet 4/1 detail
TCP Statistics for port GigabitEthernet4/1
TCP send stats
  355 segments, 37760 bytes
  222 data, 130 ack only packets
  3 control (SYN/FIN/RST), 0 probes, 0 window updates
  0 segments retransmitted, 0 bytes
  0 retransmitted while on ethernet send queue, 0 packets split
  0 delayed acks sent
TCP receive stats
  163 segments, 114 data packets in sequence, 6512 bytes in sequence
  0 predicted ack, 10 predicted data
  0 bad checksum, 0 multi/broadcast, 0 bad offset
  0 no memory drops, 0 short segments
  0 duplicate bytes, 0 duplicate packets
  0 partial duplicate bytes, 0 partial duplicate packets
  0 out-of-order bytes, 1 out-of-order packets
  0 packet after window, 0 bytes after window
  0 packets after close
  121 acks, 37764 ack bytes, 0 ack toomuch, 4 duplicate acks
  0 ack packets left of snd_una, 0 non-4 byte aligned packets
  8 window updates, 0 window probe
  30 pcb hash miss, 0 no port, 0 bad SYN, 0 paws drops
TCP Connection Stats
  0 attempts, 3 accepts, 3 established
  3 closed, 2 drops, 0 conn drops
  0 drop in retransmit timeout, 1 drop in keepalive timeout
  0 drop in persist drops, 0 connections drained
TCP Miscellaneous Stats
  115 segments timed, 121 rtt updated
  0 retransmit timeout, 0 persist timeout
  12 keepalive timeout, 11 keepalive probes
TCP SACK Stats
  0 recovery episodes, 0 data packets, 0 data bytes
  0 data packets retransmitted, 0 data bytes retransmitted
  0 connections closed, 0 retransmit timeouts
TCP SYN Cache Stats
  15 entries, 3 connections completed, 0 entries timed out
  0 dropped due to overflow, 12 dropped due to RST
  0 dropped due to ICMP unreach, 0 dropped due to bucket overflow
  0 abort due to no memory, 0 duplicate SYN, 0 no-route SYN drop
```

```

0 hash collisions, 0 retransmitted
TCP Active Connections
Local Address      Remote Address    State      Send-Q   Recv-Q
0.0.0.0:3260      0.0.0.0:0        LISTEN    0        0

```

Use the **show ips stats icmp interface gigabitethernet** to display and verify IP statistics. This command takes the main Ethernet interface as a parameter and returns the ICMP statistics for that interface. See [Example 22-13](#).

#### Example 22-13 Displays ICMP Statistics

```

switch# show ips stats icmp interface gigabitethernet 4/1
ICMP Statistics for port GigabitEthernet4/1
  5 ICMP messages received
  0 ICMP messages dropped due to errors
ICMP input histogram
  5 echo request
ICMP output histogram
  5 echo reply

```

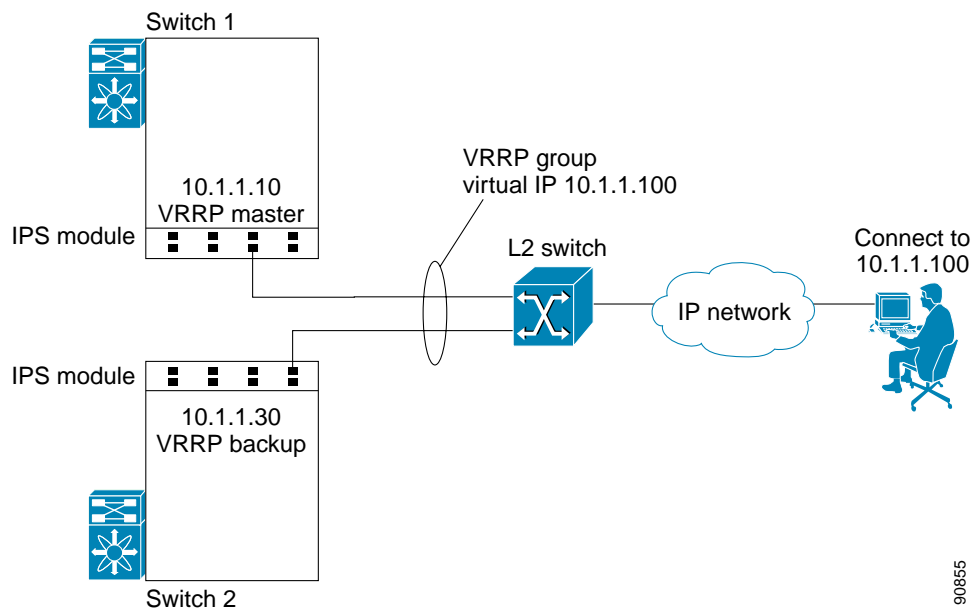
## Gigabit Ethernet High Availability

Virtual Router Redundancy Protocol (VRRP) and Ethernet PortChannels are two Gigabit Ethernet features that provide high availability for iSCSI and FCIP services.

### VRRP for iSCSI and FCIP Services

VRRP provides a redundant alternate path to the Gigabit Ethernet port for iSCSI and FCIP services. VRRP provides IP address fail over protection to an alternate Gigabit Ethernet interface so the IP address is always available (see [Figure 22-4](#)).

Figure 22-4 VRRP Scenario



90855

In [Figure 22-4](#), all members of the VRRP group must be IP storage Gigabit Ethernet ports. VRRP group members can be one or more of the following interfaces:

- One or more interfaces in the same IPS module
- Interfaces across IPS modules in one switch
- Interfaces across IPS modules in different switches
- Gigabit Ethernet subinterfaces
- Ethernet PortChannels
- Subinterfaces

See the [“Configuring VRRP” section on page 20-19](#).

## Configuring VRRP for Gigabit Ethernet Interfaces



**Note** The VRRP **preempt** option is not supported on IP storage Gigabit Ethernet interfaces. However, if the virtual IP address is also the IP address for the interface, then preemption is implicitly applied.

To configure VRRP for Gigabit Ethernet interfaces, follow these steps:

	Command	Purpose
Step 1	switch1# <b>config terminal</b> switch1(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface</b> <b>gigabitethernet 2/2</b> switch(config-if)#	Enters the interface configuration mode on the Gigabit Ethernet interface (slot 2, port 2).
Step 3	switch(config-if)# <b>ip address</b> <b>10.1.1.10 255.255.255.0</b>	Enters the IP address (10.1.1.10) and IP mask (255.255.255.0) for the Gigabit Ethernet interface.
Step 4	switch(config-if)# <b>no shutdown</b>	Enables the selected interface.
Step 5	switch(config-if)# <b>vrrp 100</b> switch(config-if-vrrp)	Creates a VR ID 100.
Step 6	switch(config-if-vrrp)# <b>address</b> <b>10.1.1.100</b>	Configures the virtual IP address (10.1.1.100) for the selected VRRP group (identified by the VR ID).  <b>Note</b> The virtual IP address must be in the same subnet as the IP address of the Gigabit Ethernet interface. All members of the VRRP group must configure the same virtual IP address.
Step 7	switch(config-if-vrrp)# <b>priority 10</b>	Configures the priority for the selected interface within this VRRP group.  <b>Note</b> The interface with the highest priority is selected as the master.
Step 8	switch(config-if-vrrp)# <b>no shutdown</b>	Enables the VRRP protocol on the selected interface.

## Ethernet PortChannel Aggregation

Ethernet PortChannels refer to the aggregation of multiple physical Gigabit Ethernet interfaces into one logical Ethernet interface to provide link redundancy and, in some cases, higher aggregated bandwidth and load balancing.

The data traffic from one TCP connection always travels on the same physical links. An Ethernet switch connecting to the MDS Gigabit Ethernet port can implement load balancing based on the IP address, IP address and UDP/TCP port number, or MAC address.

**Note**

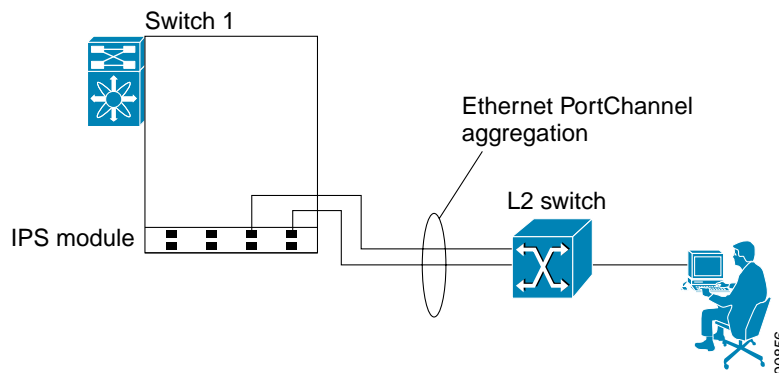
The Cisco Ethernet switch's PortChannel should be configured as a static PortChannel, and not the default 802.3ad protocol.

Ethernet PortChannels can only aggregate two physical interfaces that are adjacent to each other on a given IPS module (see [Figure 22-5](#)).

**Note**

PortChannel members must be one of these combinations: ports 1-2, ports 3-4, ports 5-6, or ports 7-8.

**Figure 22-5 Ethernet PortChannel Scenario**



In [Figure 22-5](#), Gigabit Ethernet ports 3 and 4 in slot 9 are aggregated into an Ethernet PortChannel.

**Note**

All FCIP data traffic for one FCIP link is carried on one TCP connection. Consequently, the aggregated bandwidth is 1 Gbps for that FCIP link.

**Note**

PortChannel interfaces provide configuration options for both Gigabit Ethernet and Fibre Channel. However, based on the PortChannel membership, only Gigabit Ethernet parameters or Fibre Channel parameters are applicable.

## Configuring Ethernet PortChannels

**Note**

Gigabit Ethernet interfaces cannot be added to a PortChannel if one of the following cases apply:

- The interface already has an IP address assigned.
- The subinterfaces are configured on that interface.

The PortChannel configuration specified in [Chapter 12, “Configuring PortChannels”](#) also applies to Ethernet PortChannel configurations.

To configure Ethernet PortChannels, follow these steps:



	Command	Purpose
Step 1	switch1# <b>config terminal</b> switch1(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface port-channel 10</b> switch(config-if)#	Configures the specified PortChannel (10).
Step 3	switch(config-if)# <b>ip address 10.1.1.1</b> <b>255.255.255.0</b>	Enters the IP address (10.1.1.1) and IP mask (255.255.255.0) for the PortChannel. <b>Note</b> A PortChannel does not have any members when first created.
Step 4	switch(config-if)# <b>no shutdown</b>	Enables the interface.
Step 5	switch(config)# <b>interface gigabitethernet 9/3</b> switch(config-if)#	Configures the specified Gigabit Ethernet interface (slot 9, port 3).
Step 6	switch(config-if)# <b>channel-group 10</b> gigabitethernet 9/3 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 switch(config-if)#	Adds Gigabit Ethernet interfaces 9/3 to channel group 10. If channel group 10 does not exist, it is created. The port is shut down.
Step 7	switch(config-if)# <b>no shutdown</b>	Enables the selected interface.
Step 8	switch(config)# <b>interface gigabitethernet 9/4</b> switch(config-if)#	Configures the specified Gigabit Ethernet interface (slot 9, port 4).
Step 9	switch(config-if)# <b>channel-group 10</b> gigabitethernet 9/4 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	Adds Gigabit Ethernet interfaces 9/4 to channel group 10. The port is shut down.
Step 10	switch(config-if)# <b>no shutdown</b>	Enables the selected interface.

## Configuring CDP

The Cisco Discovery Protocol (CDP) is supported on the management Ethernet interface on the supervisor module and the Gigabit Ethernet interface on the IPS module.

See the “[CDP Configuration](#)” section on page 4-36.

## IPS Core Dumps

IPS core dumps are different from the system’s kernel core dumps for other modules. When the IPS module’s operating system (OS) unexpectedly resets, it is sometimes useful to obtain a copy of the memory image (called a IPS core dump) to identify the cause of the reset. Under that condition, the IPS module sends the core dump to the supervisor module for storage. Core dumps take up significant space and hence the level of what gets stored can be configured using one of the following options:

- Partial core dumps (default)—Each partial core dump consists of four parts (four files).
- Full core dumps—Each full core dump consists of 75 parts (75 files). This dump cannot be saved on the supervisor module due to its large space requirement.

## Configuring IPS Core Dumps

To configure IPS core dumps on the IPS module, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>ips core dump full</b> ips core dump full' successfully set for module 9	Configures a dump of the full core generation for the IPS module in slot 9.
	switch(config)# <b>no ips core dump full</b> ips core dump partial' successfully set for module 9	Configures a dump of the partial core generation for the IPS module in slot 9.

# Configuring FCIP

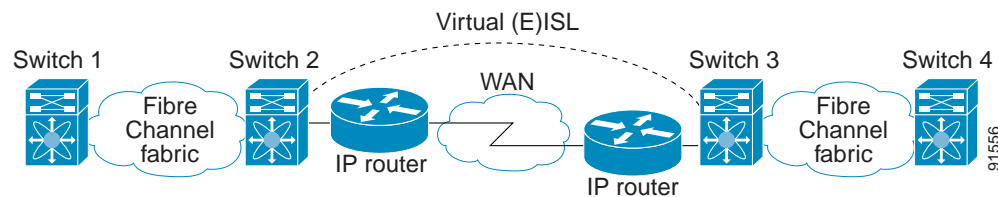
This section includes the following topics:

- [About FCIP, page 22-19](#)
- [Basic FCIP Configuration, page 22-22](#)
- [Advanced FCIP Profile Configuration, page 22-24](#)
- [Advanced FCIP Interface Configuration, page 22-30](#)
- [E Port Configurations, page 22-36](#)
- [Displaying FCIP Information, page 22-38](#)
- [FCIP High Availability, page 22-41](#)
- [Ethernet PortChannels and Fibre Channel PortChannels, page 22-44](#)

## About FCIP

The Fibre Channel over IP Protocol (FCIP) is a tunneling protocol that connects geographically distributed Fibre Channel storage area networks (SAN islands) transparently over IP local area networks (LANs), metropolitan area networks (MANs), and wide area networks (WANs). See [Figure 22-6](#).

**Figure 22-6 Fibre Channel SANs Connected by FCIP**



FCIP uses TCP as a network layer transport.



### Note

For more information about FCIP protocols, refer to the IETF standards for IP storage at <http://www.ietf.org>. Also refer to Fibre Channel standards for switch backbone connection at <http://www.t11.org> (see FC-BB-2).

To configure the IPS module for FCIP, you should have a basic understanding of the following concepts:

- [FCIP and VE Ports, page 22-20](#)
- [FCIP Links, page 22-20](#)
- [FCIP Profiles, page 22-21](#)
- [FCIP Interfaces, page 22-21](#)

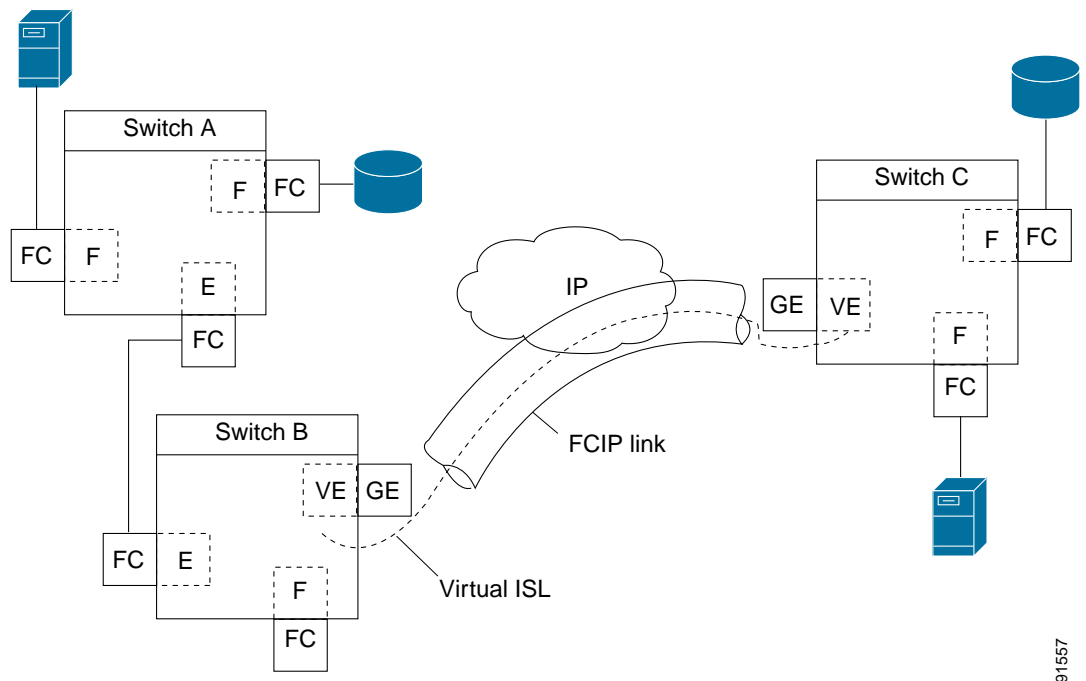
## FCIP and VE Ports

Figure 22-7 describes the internal model of FCIP with respect to Fibre Channel Inter-Switch Links (ISLs) and Cisco's enhanced ISLs (EISLs).

FCIP virtual E (VE) ports behave exactly like standard Fibre Channel E ports, except that the transport in this case is FCIP instead of Fibre Channel. The only requirement is for the other end of the VE port to be another VE port.

A virtual ISL is established over a FCIP link and transports Fibre Channel traffic. Each associated virtual ISL looks like a Fibre Channel ISL with either an E port or a TE port at each end (see Figure 22-7).

Figure 22-7 FCIP Links and Virtual ISLs



91557

See the “E Port” section on page 10-3.

## FCIP Links

FCIP links consist of one or more TCP connections between two FCIP link endpoints. Each link carries encapsulated Fibre Channel frames.

When the FCIP link comes up, the VE ports at both ends of the FCIP link create a virtual Fibre Channel (E)ISL and initiate the E port protocol to bring up the (E)ISL.

By default, the FCIP feature on any Cisco MDS 9000 Family switch creates two TCP connections for each FCIP link.

- One connection is used for data frames.
- The other connection is used only for Fibre Channel control frames, that is, switch-to-switch protocol frames (all Class F). This arrangement provides low latency for all control frames.

To enable FCIP on the IPS module, a FCIP profile and FCIP interface (interface FCIP) must be configured.

The FCIP link is established between two peers, the VE port initialization behavior is identical to a normal E port. This behavior is independent of the link being FCIP or pure Fibre Channel, and is based on the E port discovery process (ELP, ESC).

Once the FCIP link is established, the VE port behavior is identical to E port behavior for all inter-switch communication (including domain management, zones, and VSANs). At the Fibre Channel layer, all VE and E port operations are identical.

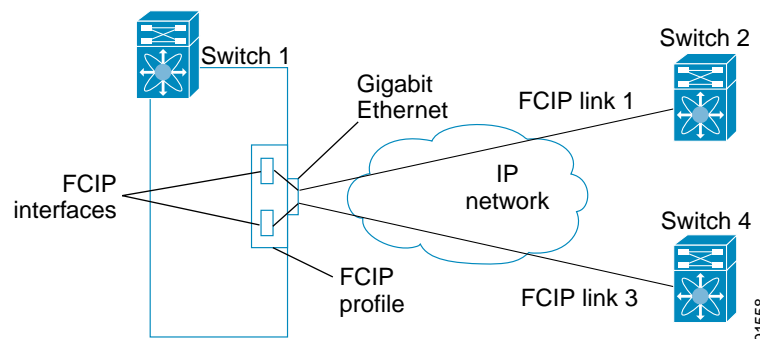
## FCIP Profiles

The FCIP profile contains information about local IP address and TCP parameters. The profile defines the following information:

- The local connection points (IP address and TCP port number).
- The behavior of the underlying TCP connections for all FCIP links that use this profile.

The FCIP profile's local IP address determines the Gigabit Ethernet port where the FCIP links terminate (see [Figure 22-8](#)).

**Figure 22-8 FCIP Profile and FCIP Links**



## FCIP Interfaces

The FCIP interface is the local endpoints of the FCIP link and a VE port interface. All the FCIP and E port parameters are configured in context to the FCIP interface.

The FCIP parameters consist of the following:

- The FCIP profile determines which Gigabit Ethernet port initiates the FCIP links and defines the TCP connection behavior.
- Peer information.
- Number of TCP connections for the FCIP link.
- E port parameters—trunking mode and trunk allowed VSAN list.

## Enabling FCIP

To begin configuring the FCIP feature, you must explicitly enable FCIP on the required switches in the fabric. By default, this feature is disabled in all switches in the Cisco MDS 9000 Family.

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

To enable FCIP on any participating switch, follow these steps:

	Command	Purpose
Step 1	switch# <b>conf t</b>	Enters configuration mode.
Step 2	switch(config)# <b>fcip enable</b>	Enables FCIP on that switch.
	switch(config)# <b>no fcip enable</b>	Disables (default) FCIP on that switch.

## Basic FCIP Configuration

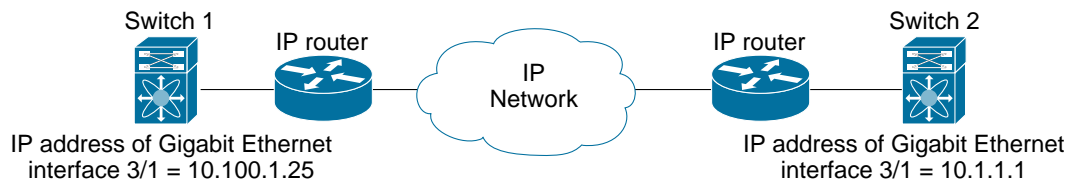
To configure a FCIP link, follow these steps on both switches:

- Step 1 Configure the Gigabit Ethernet interface.
- Step 2 Create a FCIP profile, and then assign the Gigabit Ethernet interface's IP address to the profile.
- Step 3 Create a FCIP interface, and then assign the profile to the interface.
- Step 4 Configure the peer IP address for the FCIP interface.
- Step 5 Enable the interface.

## Creating FCIP Profiles

You must assign a local IP address of a Gigabit Ethernet interface or subinterface to the FCIP profile to create a FCIP profile (see [Figure 22-9](#)).

**Figure 22-9 Assigning Profiles to Each Gigabit Ethernet Interface**



To create a FCIP profile in switch 1, follow these steps:

	Command	Purpose
Step 1	switch1# <b>conf terminal</b> switch1(config)#	Enters configuration mode.

	Command	Purpose
Step 2	switch1(config)# <b>fcip profile 10</b> switch1(config-profile)#	Creates a profile for the FCIP connection. The valid range is from 1 to 255.
Step 3	switch1(config-profile)# <b>ip address 10.100.1.25</b>	Associates the profile (10) with the local IP address of the Gigabit Ethernet interface (3/1).

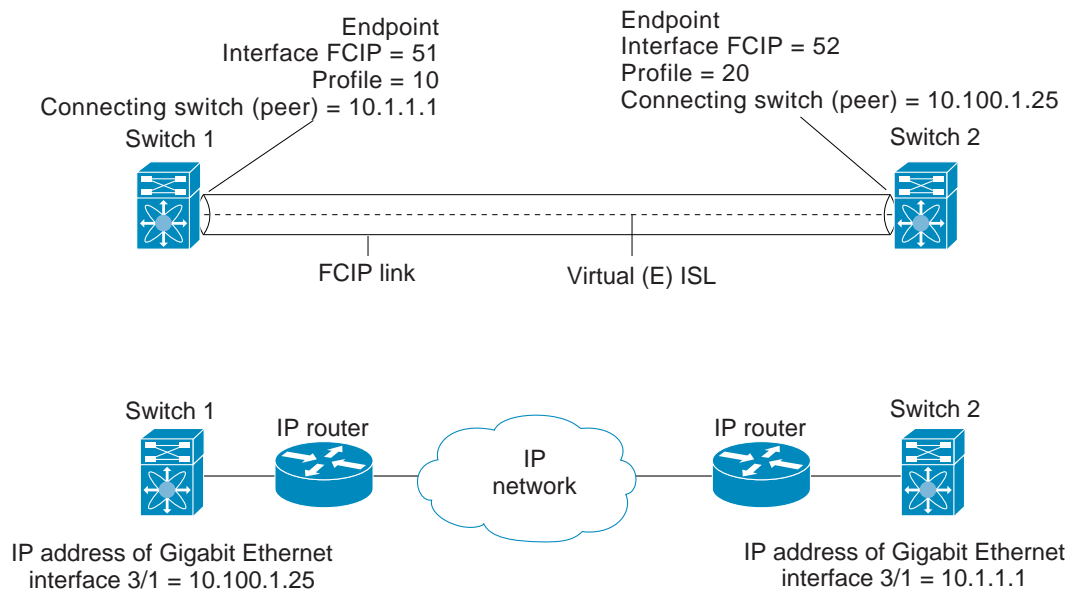
To assign FCIP profile in switch 2, follow these steps:

	Command	Purpose
Step 1	switch2# <b>config terminal</b> switch2(config)#	Enters configuration mode.
Step 2	switch2(config)# <b>fcip profile 20</b> switch2(config-profile)#	Creates a profile for the FCIP connection.
Step 3	switch2(config-profile)# <b>ip address 10.1.1.1</b>	Associates the profile (20) with the local IP address of the Gigabit Ethernet interface.

## Creating FCIP Links

When two FCIP link endpoints are created, a FCIP link is established between the two IPS modules. To create a FCIP link, assign a profile to the FCIP interface and configure the peer information. The peer IP switch information initiates (creates) a FCIP link to that peer switch (see [Figure 22-10](#)).

**Figure 22-10 Assigning Profiles to Each Gigabit Ethernet Interface**



91562

To create a FCIP link endpoints in switch 1, follow these steps:

	Command	Purpose
Step 1	switch1# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch1(config)# <b>interface fcip 51</b> switch1(config-if)#	Creates a FCIP interface (51).
Step 3	switch1(config-if)# <b>use-profile 10</b>	Assigns the profile (10) to the FCIP interface.
Step 4	switch1(config-if)# <b>peer-info ipaddr 10.1.1.1</b>	Assigns the peer IP address information (10.1.1.1 for switch 2) to the FCIP interface.
Step 5	switch1(config-if)# <b>no shutdown</b>	Enables the interface.

To create a FCIP link endpoints in switch 2, follow these steps:

	Command	Purpose
Step 1	switch2# <b>config terminal</b> switch2(config)#	Enters configuration mode.
Step 2	switch2(config)# <b>interface fcip 52</b> switch2(config-if)#	Creates a FCIP interface (52).
Step 3	switch2(config-if)# <b>use-profile 20</b>	Binds the profile (20) to the FCIP interface.
Step 4	switch2(config-if)# <b>peer-info ip address 10.100.1.25</b>	Assigns the peer IP address information (10.100.1.25 for switch 1) to the FCIP interface.
Step 5	switch1(config-if)# <b>no shutdown</b>	Enables the interface.

## Advanced FCIP Profile Configuration

A basic FCIP configuration uses the local IP address to configure the FCIP profile. In addition to the local IP address and the local port, you can specify other TCP parameters as part of the FCIP profile configuration.

- [Configuring TCP Listener Ports, page 22-25](#)
- [Configuring TCP Parameters, page 22-25](#)

FCIP configuration options can be accessed from the `switch(config-profile)#` submode prompt.

To enter the `switch(config-profile)#` prompt, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>fcip profile 20</b> switch(config-profile)#	Creates the profile (if it does not already exist). The valid range is from 1 to 255.



## Configuring TCP Listener Ports

The default TCP port for FCIP is 3225.

You can change this port using the **port** command.

To change the default FCIP port number (3225), follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# port 5000</code>	Associates the profile with the local port number (5000).
	<code>switch(config-profile)# no port</code>	Reverts to the default 3225 port.

## Configuring TCP Parameters

You can control TCP behavior in a switch by configuring the following TCP parameters.

- [Minimum Retransmit Timeout, page 22-25](#)
- [Keepalive Timeout, page 22-25](#)
- [Maximum Retransmissions, page 22-26](#)
- [Path MTU, page 22-26](#)
- [Selective acknowledgment, page 22-27](#)
- [Window Management, page 22-27](#)
- [Buffer Size, page 22-28](#)
- [Quality of Service, page 22-28](#)
- [Monitoring Window Congestion, page 22-29](#)
- [Estimating Maximum Delay Jitter, page 22-29](#)

### Minimum Retransmit Timeout

You can control the minimum amount of time TCP waits before retransmitting. By default, this value is 200 milliseconds (ms).

To configure the minimum retransmit time, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# tcp min-retransmit-time 500</code>	Specifies the minimum TCP retransmit time for the TCP connection to be 500 ms. The default is 200 ms and the range is from 200 to 5000 ms.
	<code>switch(config-profile)# no tcp min-retransmit-time 500</code>	Reverts the minimum TCP retransmit time to the factory default of 200 ms.

### Keepalive Timeout

You can configure the interval that the TCP connection uses to verify that the FCIP link is functioning. This ensures that a FCIP link failure is detected quickly even when there is no traffic.

If the TCP connection is idle for more than the specified time, then keepalive timeout packets are sent to ensure that the connection is active. This command can be used to tune the time taken to detect FCIP link failures.

The first interval during which the connection is idle is 60 seconds (default). When the connection is idle for 60 seconds, eight keepalive probes are sent at 1-second intervals. If no response is received for these eight probes and the connection remains idle throughout, that FCIP link is automatically closed.

**Note**

Only the first interval (during which the connection is idle) can be changed from the default of 60 seconds. This interval is identified using the **tcp keepalive-timeout** parameter. The valid range is from 1 to 7200 seconds.

To configure the keepalive timeout, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# tcp keepalive-timeout 120</code>	Specifies the keepalive timeout interval for the TCP connection in seconds (120). The default is 60 seconds. The range is from 1 to 7200 seconds.
	<code>switch(config-profile)# no tcp keepalive-timeout 120</code>	Reverts the keepalive timeout to 60 seconds.

## Maximum Retransmissions

The **tcp max-retransmissions** parameter specifies the maximum number of times a packet is retransmitted before TCP decides to close the connection.

To configure maximum retransmissions, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# tcp max-retransmissions 6</code>	Specifies the maximum number of retransmissions (6). The default is 4 and the range is from 1 to 8 retransmissions.
	<code>switch(config-profile)# no tcp max-retransmissions 6</code>	Reverts to the default of 4 retransmissions.

## Path MTU

Path MTU (PMTU) is the minimum MTU on the IP network between the two endpoints of the FCIP link. PMTU discovery is a mechanism by which TCP learns of the PMTU dynamically and adjusts the maximum TCP segment accordingly (RFC 1191).

By default, PMTU discovery is enabled on all switches with a default timeout of 3600 seconds. If TCP reduces the size of the maximum segment because of PMTU change, the reset-timeout specifies the time after which TCP tries the original MTU.

To configure PMTU, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# no tcp pmtu-enable</code>	Disables PMTU discovery.
	<code>switch(config-profile)# tcp pmtu-enable</code>	Enables (default) PMTU discovery with the default value of 3600 seconds.
	<code>switch(config-profile)# tcp pmtu-enable reset-timeout 90</code>	Specifies the PMTU reset timeout to 90 seconds. The default is 3600 seconds and the range is from 60 to 3600 seconds.
	<code>switch(config-profile)# no tcp pmtu-enable reset-timeout 600</code>	Leaves the PMTU in an enabled state but changes the timeout to the default of 3600 seconds.

## Selective acknowledgment

TCP may experience poor performance when multiple packets are lost within one window. With the limited information available from cumulative acknowledgments, a TCP sender can only learn about a single lost packet per round trip time. A selective acknowledgment (SACK) mechanism helps overcome the limitations of multiple lost packets during a TCP transmission.

The receiving TCP sends back SACK advertisements to the sender. The sender can then retransmit only the missing data segments. By default, SACK is enabled on Cisco MDS 9000 Family switches.

To configure SACK, follow these steps:

	Command	Purpose
Step 1	switch(config-profile)# <b>no tcp sack-enable</b>	Disables SACK.
	switch(config-profile)# <b>tcp sack-enable</b>	Enables SACK (default).

## Window Management

You can compute the optimal TCP window size using the **max-bandwidth** parameter, the **min-available-bandwidth** parameter, and the dynamically measured round-trip-time (RTT).



### Note

The configured **round-trip-time** parameter determines the window scaling factor of the TCP connection. This parameter is only an approximation. The measured RTT value overrides the **round-trip-time** parameter for window management. If the configured **round-trip-time** is too small compared to the measured RTT, then the link may not be fully utilized due to the window scaling factor being too small.

The interaction and the resulting TCP behavior is outlined below:

- If the average rate of the fc traffic over the preceding RTT is less than the min-available-bandwidth \* RTT, every FC burst is sent immediately at the min-available-bandwidth rate, provided no TCP drops occur.
- If the average rate of the FC traffic is greater than min-available-bandwidth \* RTT, but less than max-bandwidth \* RTT, then if the FC traffic is transmitted in burst sizes smaller than the configured CWM value all the bursts are sent immediately by FCIP at the max-bandwidth rate.
- If the average rate of the FC traffic is larger than the min-available-bandwidth \* RTT and the burst size is greater than the CWM value, some traffic is not sent immediately.

The software uses standard TCP rules to increase the window beyond the one required to maintain the min-available-bandwidth to reach the max-bandwidth.

The FCIP defaults are max-bandwidth = 1G, min-available-bandwidth = 500 Mbps, and round-trip-time = 1 ms.

The **min-available-bandwidth** parameter and the measured round-trip-time together determine the threshold below which TCP aggressively maintains a window size sufficient to transmit at min-available-bandwidth.

The **maximum-bandwidth** parameter and the measured round-trip-time together determine the maximum window size.

To configure window management, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# tcp max-bandwidth-mbps 900 min-available-bandwidth-mbps 300 round-trip-time-ms 10</code>	Configures the maximum available bandwidth at 900 Mbps, the minimum slow start threshold at 300 Mbps, and the round trip time at 10 ms.
	<code>switch(config-profile)# no tcp max-bandwidth-mbps 900 min-available-bandwidth-mbps 300 round-trip-time-ms 10</code>	Reverts to the factory defaults. The FCIP defaults are max-bandwidth at 1G, min-available-bandwidth at 500 Mbps and round-trip-time at 1 ms.
	<code>switch(config-profile)# tcp max-bandwidth-kbps 2000 min-available-bandwidth-kbps 2000 round-trip-time-us 200</code>	Configures the maximum available bandwidth at 2000 Kbps, the minimum slow start threshold at 2000 Kbps, and the round-trip-time at 200 ms.

## Buffer Size

You can define the required additional buffering—beyond the normal send window size—that TCP allows before flow controlling the switch's egress path for the FCIP interface. The default FCIP buffer size is 0 KB.

To set the buffer size, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# tcp send-buffer-size 5000</code>	Configure the advertised buffer size to 5000 KB. The valid range is from 0 to 8192 KB.
	<code>switch(config-profile)# no tcp send-buffer-size 5000</code>	Reverts the switch to its factory default.

## Quality of Service

The Quality of Service (QoS) parameter specifies the differentiated services code point (DSCP) value to mark all IP packets (type of service—TOS field in the IP header).

- The control DSCP value applies to all FCIP frames in the control TCP connection.
- The data DSCP value applies to all FCIP frames in the data connection.

If the FCIP link has only one TCP connection, that data DSCP value is applied to all packets in that connection.

To set the QoS values, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# tcp qos control 3 data 5</code>	Configures the control TCP connection and data connection to mark all packets on that DSCP value.
	<code>switch(config-profile)# no tcp qos control 3 data 5</code>	Reverts the switch to its factory default (marks all packets with DSCP value 0).

## Monitoring Window Congestion

The congestion window monitoring (CWM) parameter determines the maximum burst size allowed after an idle period.

- If the FC traffic burst is smaller than the configured CWM value, every packet is sent immediately, provided that no TCP drops were detected in the previous RTT.
- If FC traffic burst is larger than the configured CWM value, the excess packets are sent during succeeding RTTs.

By default, this parameter is enabled and the default burst size is 10 KB.



### Tip

We recommend that this feature remain enabled to realize optimal performance. Increasing the CWM burst size can result in more packet drops in the IP network, impacting TCP performance. Only if the IP network has sufficient buffering, try increasing the CWM burst size beyond the default to achieve lower transmit latency.

To change the CWM defaults, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# no tcp cwm</code>	Disables congestion monitoring.
	<code>switch(config-profile)# tcp cwm</code>	Enables congestion monitoring and sets the burst size to its default size.
	<code>switch(config-profile)# tcp cwm burstsize 30</code>	Changes the burst size to 30 KB. The valid range is from 10 to 100 KB.
	<code>switch(config-profile)# no tcp cwm burstsize 25</code>	Leaves the CWM feature in an enabled state but changes the burst size to its factory default.

## Estimating Maximum Delay Jitter

As of Cisco MDS SAN-OS Release 1.3(4), you can configure the maximum estimated delay jitter in microseconds by the packet sender. The estimated variation should not include network queuing delay. By default, this parameter is enabled in Cisco MDS switches when IPS modules are present.

The default value is 100 microseconds for FCIP interfaces and 500 microseconds for iSCSI interfaces. The valid range is from 0 to 10000 microseconds.

To configure the maximum jitter value, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-profile)# no tcp max-jitter</code>	Disables delay jitter estimation.
	<code>switch(config-profile)# tcp max-jitter</code>	Enables the delay jitter feature and sets the time to its factory default.
	<code>switch(config-profile)# tcp max-jitter 300</code>	Changes the time to 300 microseconds. The valid range is from 0 to 10000 microseconds.
	<code>switch(config-profile)# no tcp max-jitter 2500</code>	Leaves the delay jitter feature in an enabled state but changes the time to its factory default.

## Advanced FCIP Interface Configuration

You can establish connection to a peer by configuring one or more of the following options for the FCIP interface. To do so, you must first create the interface and enter the `config-if` submode.

- [Configuring Peers, page 22-30](#)
- [Configuring Active Connections, page 22-31](#)
- [Configuring the Number of TCP Connections, page 22-32](#)
- [Enabling Timestamps, page 22-32](#)
- [B Port Interoperability Mode, page 22-34](#)
- [Configuring FCIP Write Acceleration, page 22-37](#)

To enter the `config-if` submode, follow these steps:

	Command	Purpose
Step 1	<code>switch# config terminal</code>	Enters configuration mode.
Step 2	<code>switch(config)# interface fcip 100</code>	Creates a FCIP interface (100).

## Configuring Peers

To establish a FCIP link with the peer, you can use one of two options:

- Peer IP address—Configures both ends of the FCIP link. Optionally, you can also use the peer TCP port along with the IP address.
- Special frames—Configures one end of the FCIP link when security gateways are present in the IP network. Optionally, you can also use the port and profile ID along with the IP address.

### Peer IP Address

The basic FCIP configuration uses the peer's IP address to configure the peer information. You can also specify the peer's port number to configure the peer information. If you do not specify a port, the default 3225 port number is used to establish connection.

To assign the peer information based on the IP address, port number, or a profile ID, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-if)# peer-info ipaddr 10.1.1.1</code>	Assigns an IP address to configure the peer information. Because no port is specified, the default port number, 3225, is used.
	<code>switch(config-if)# no peer-info ipaddr 10.10.1.1</code>	Deletes the assigned peer port information.
Step 2	<code>switch(config-if)# peer-info ipaddr 10.1.1.1 port 3000</code>	Assigns the IP address and sets the peer TCP port to 3000. The valid port number range is from 0 to 65535.
	<code>switch(config-if)# no peer-info ipaddr 10.1.1.1 port 2000</code>	Deletes the assigned peer port information.
Step 3	<code>switch(config-if)# no shutdown</code>	Enables the interface.

## Special Frames

You can alternatively establish a FCIP link with a peer using an optional protocol called special frames. When special frames are enabled, the peer IP address (and optionally the port or the profile ID) only needs to be configured on one end of the link. Once the connection is established, a special frame is exchanged to discover and authenticate the link.

By default, the special frame feature is disabled.



### Note

Refer to the Fibre Channel IP standards for further information on special frames.



### Tip

Special frame negotiation provides an additional authentication security mechanism because the link validates the WWN of the peer switch.

You can enable or disable the **special-frame** option. On the peer side, the **special-frame** option must be enabled to establish the FCIP link.

To enable special frames, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-if)# special-frame peer-wwn 12:12:34:45:ab:bc:cd:00</code>	Enables special frames and sets the peer WWN as specified.  <b>Note</b> The peer WWN is the WWN of the peer switch. Use the <b>show wwn switch</b> command to obtain the peer WWN.
	<code>switch(config-if)# no special-frame peer-wwn 12:12:34:45:ab:bc:cd:00</code>	Disables special frames (default).
Step 2	<code>switch(config-if)# special-frame peer-wwn 12:12:34:45:ab:bc:cd:00 peer profile-id 155</code>	Enables special frames and sets the peer WWN as specified by the profile ID (155).
	<code>switch(config-if)# no special-frame peer-wwn 12:12:34:45:ab:bc:cd:00 peer profile-id 155</code>	Disables special frames (default).
Step 3	<code>switch(config-if)# no shutdown</code>	Enables the interface.

## Configuring Active Connections

You can configure the required mode for initiating an IP connection. By default, active mode is enabled to actively attempt an IP connection.

If you enable the passive mode, the switch does not initiate a TCP connection and merely waits for the peer to connect to it.



### Note

Ensure that both ends of the FCIP link are not configured as passive mode. If both ends are configured as passive, the connection is not initiated.

Use the **passive-mode** option to configure the required mode for initiating an IP connection.

To enable the passive mode, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-if)# <b>passive-mode</b></code>	Enables passive mode while attempting a TCP connection.
	<code>switch(config-if)# <b>no passive-mode</b></code>	Reverts to the factory set default of using the active mode while attempting the TCP connection.
Step 2	<code>switch(config-if)# <b>no shutdown</b></code>	Enables the interface.

## Configuring the Number of TCP Connections

You can specify the number of TCP connections from a FCIP link. By default, the switch tries two (2) TCP connections for each FCIP link. You can configure one or two TCP connections. For example, the Cisco PA-FC-1G Fibre Channel port adapter, which has only one (1) TCP connection, interoperates with any switch in the Cisco MDS 9000 Family. One TCP connection is within the specified limit. If the peer initiates one TCP connection, and your MDS switch is configured for two TCP connections, then the software handles it gracefully and moves on with just one connection.

Use the **tcp-connection** option to specify the number of TCP connections from a FCIP link. You can change the configuration on the switch using the **tcp-connection 1** command.

To specify the TCP connection attempts, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-if)# <b>tcp-connection 1</b></code>	Specifies the number of TCP connections. Two (2) is the default and the maximum number of TCP connection attempts.
	<code>switch(config-if)# <b>no tcp-connection 1</b></code>	Reverts to the factory set default of two attempts.
Step 2	<code>switch(config-if)# <b>no shutdown</b></code>	Enables the interface.

## Enabling Timestamps

You can instruct the switch to discard packets that are outside the specified time. By default, this option is disabled in all switches in the Cisco MDS 9000 Family.

The **acceptable-diff** option specifies the time range within which packets can be accepted. If the packet arrived within the range specified by this option, the packet is accepted. Otherwise, it is dropped. By default if a packet arrives within a 1000 millisecond interval (+ or -1000 ms), that packet is accepted.

Use the **time-stamp** option to enable or disable FCIP timestamps on a packet.



### Note

If the **time-stamp** option is enabled, be sure to configure NTP on both switches (see the “[NTP Configuration](#)” section on page 4-18).

To enable or disable the **time-stamp** option, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-if)# <b>time-stamp</b></code> Please enable NTP with a common time source on both MDS Switches that are on either side of the FCIP link	Enables timestamp checking for received packets with a default acceptable time difference of 1000 ms.
	<code>switch(config-if)# <b>no time-stamp</b></code>	Disables (default) timestamps.

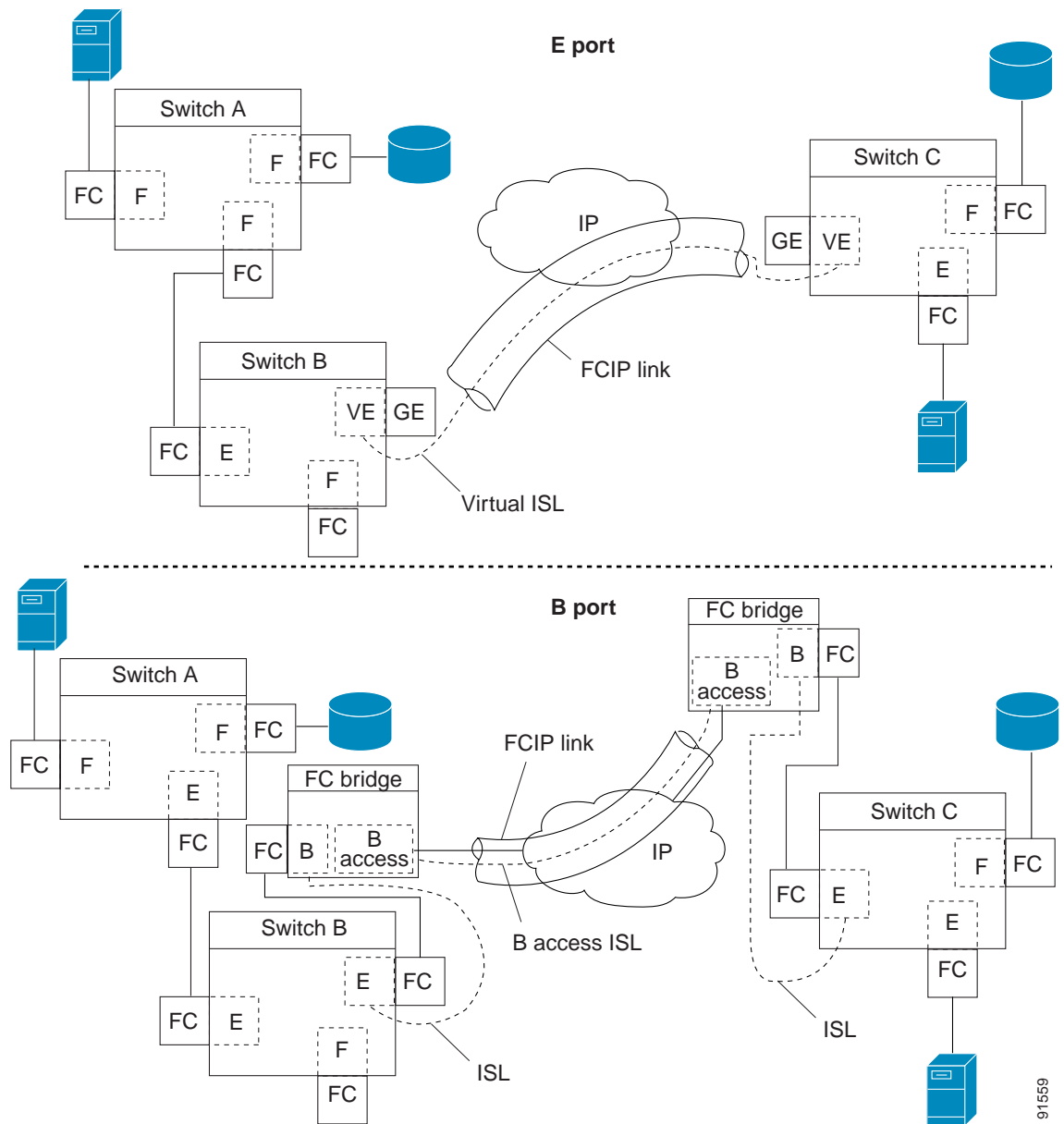


	Command	Purpose
Step 2	<code>switch(config-if)# time-stamp acceptable-diff 4000</code>	Configures the acceptable time within which a packet is accepted. The default difference is a 1000 millisecond interval from the network time. The valid range is from 500 to 10,000 ms.
	<code>switch(config-if)# no time-stamp acceptable-diff 500</code>	Deletes the configured time difference and reverts the difference to factory defaults.
Step 3	<code>switch(config-if)# no shutdown</code>	Enables the interface.

## B Port Interoperability Mode

While E ports typically interconnect Fibre Channel switches, some SAN extender devices, such as Cisco's PA-FC-1G Fibre Channel port adapter and the SN 5428-2 storage router, implement a bridge port model to connect geographically dispersed fabrics. This model uses B port as described in the T11 Standard FC-BB-2. [Figure 22-11](#) depicts a typical SAN extension over an IP network.

Figure 22-11 FCIP B Port and Fibre Channel E Port



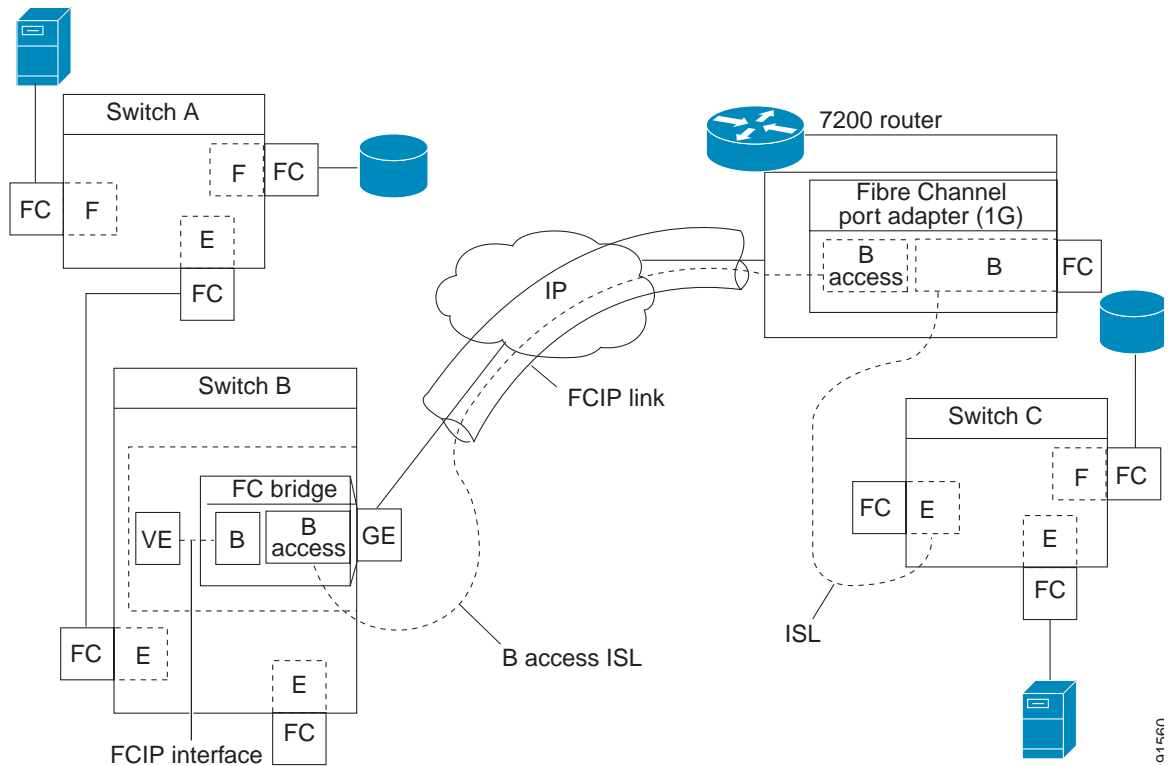
B ports bridge Fibre Channel traffic from one E port to a remote E port without participating in fabric-related activities such as principal switch election, domain ID assignment, and Fibre Channel routing (FSPF). For example, Class F traffic entering a SAN extender does not interact with the B port.

The traffic is transparently propagated (bridged) over a WAN interface before exiting the remote B port. This bridge results in both E ports exchanging Class F information that ultimately leads to normal ISL behavior such as fabric merging and routing.

FCIP links between B port SAN extenders do not exchange the same information as FCIP links between E ports, and are therefore incompatible. This is reflected by the terminology used in FC-BB-2: *while VE ports establish a virtual ISL over a FCIP link, B ports use a B access ISL*.

The IPS module supports FCIP links that originate from a B port SAN extender device by implementing the B access ISL protocol on a Gigabit Ethernet interface. Internally, the corresponding virtual B port connects to a virtual E port that completes the end-to-end E port connectivity requirement (see [Figure 22-12](#)).

**Figure 22-12 FCIP Link Terminating in a B Port Mode**



The B port feature in the IPS module allows remote B port SAN extenders to communicate directly with a Cisco MDS 9000 Family switch, therefore eliminating the need for local bridge devices.

## Configuring B Ports

When a FCIP peer is a SAN extender device that only supports Fibre Channel B ports, you need to enable the B port mode for the FCIP link. When a B port is enabled, the E port functionality is also enabled and they coexist. If the B port is disabled, the E port functionality remains enabled.

To enable B port mode, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-if)# bport</code>	Enables B port mode on the FCIP interface.
	<code>switch(config-if)# no bport</code>	Reverts to E port mode on the FCIP interface (default).
Step 2	<code>switch(config-if)# bport-keepalive</code>	Enables the reception of keepalive responses sent by a remote peer.
Step 3	<code>switch(config-if)# no bport-keepalive</code>	Disables the reception of keepalive responses sent by a remote peer (default).

## E Port Configurations

All configuration commands that apply to E ports, also apply to FCIP interfaces. The following features are also available FCIP interfaces:

- VSANs: FCIP interfaces can be a member of any VSAN.
- Trunk mode and trunk allowed VSANs can be configured
- PortChannels
  - Multiple FCIP links can be bundled into a Fibre Channel PortChannel.
  - FCIP links and Fibre Channel links cannot be combined in one PortChannel.
- FSPF
- Fibre Channel domains (fcdomains)
- Zone merge: The zone database can be imported or exported from the adjacent switch.

See [Chapter 9, “Configuring and Managing VSANs,”](#) [Chapter 11, “Configuring Trunking,”](#) [Chapter 12, “Configuring PortChannels,”](#) [Chapter 19, “Configuring Fibre Channel Routing Services and Protocols,”](#) [Chapter 24, “Configuring Domain Parameters,”](#) and [Chapter 13, “Configuring and Managing Zones.”](#)

## Configuring FCIP Write Acceleration

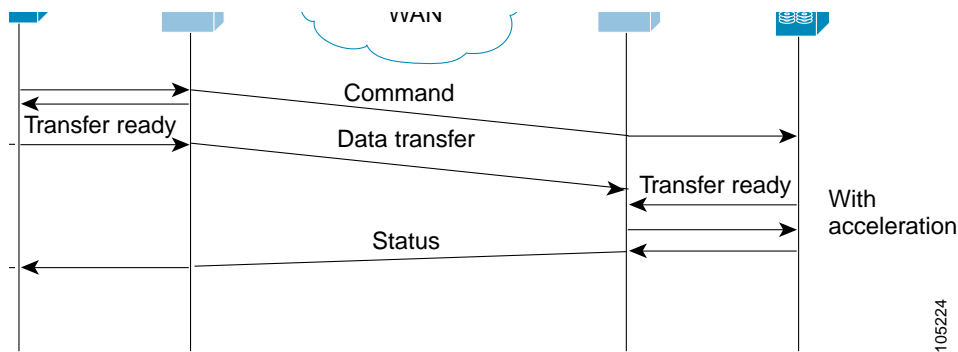
The FCIP write acceleration feature in SAN-OS 1.3(3) enables you to significantly improve application performance when storage traffic is routed over wide area networks using FCIP. When FCIP write acceleration is enabled, WAN throughput is maximized by minimizing the impact of WAN latency for the command to transfer ready acknowledgments (see [Figure 22-13](#)).



### Note

The write acceleration feature is disabled by default and must be enabled on both sides of the FCIP link. If it is only enabled on one side of the FCIP tunnel, the tunnel is not initialized.

**Figure 22-13 FCIP Link Write Acceleration**



In [Figure 22-13](#), some data sent by the host is queued on the target before the target issues a Transfer Ready. This way the actual write operation may be done in less time than the write operation without the write acceleration feature being enabled.



### Tip

FCIP write acceleration does not work if the FCIP port is part of a PortChannel or if there are multiple paths with equal weight between the initiator and the target port. Such a configuration might cause either SCSI discovery failure or broken write or read operations.



### Caution

When write acceleration is enabled in an FCIP interface, a FICON VSAN cannot be enabled in that interface. Likewise, if a FCIP interface is up in a FICON VSAN, write acceleration cannot be enabled on that interface.

To enable write acceleration, follow these steps:

	Command	Purpose
Step 1	switch1# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch1(config)# <b>interface fcip 51</b> switch1(config-if)#	Creates a FCIP interface (51).
Step 3	switch1(config-if)# <b>write-accelerator</b>	Enables write acceleration.
	switch1(config-if)# <b>no write-accelerator</b>	Disables write acceleration (default).

## Enabling FCIP Compression

The FCIP compression feature introduced in Cisco MDS SAN-OS Release 1.3(x) 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.

This feature uses the Lempel-Ziv-Stac (LZS) compression algorithm to compress packets.

The **high-throughput** mode allows faster compression but the compression ratio may be lower. The **high-comp-ratio** mode allows a higher compression ratio, but the throughput may be lower.

To enable FCIP compression, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface fcip 51</b> switch(config-if)#	Creates a FCIP interface (51).
Step 3	switch(config-if)# <b>ip-compression high-throughput</b>	Enables faster compression.
	switch(config-if)# <b>ip-compression high-comp-ratio</b>	Enables a better compression ratio.
	switch(config-if)# <b>no ip-compression</b>	Disables (default) the FCIP compression feature.

## Displaying FCIP Information

Use the **show interface** commands to view the summary, counter, description, and status of the FCIP link. Use the output of these commands to verify the administration mode, the interface status, the operational mode, the related VSAN ID, and the profile used. See Examples 22-14 to 22-19.

### Example 22-14 Displays the FCIP Interface

```
switch# show interface fcip 3
fcip3 is trunking
  Hardware is GigabitEthernet
  Port WWN is 20:ca:00:05:30:00:07:1e
  Peer port WWN is 20:ca:00:00:53:00:18:1e
  Admin port mode is auto, trunk mode is on
  Port mode is TE
  vsan is 1
  Trunk vsans (allowed active) (1,10)
  Trunk vsans (operational) (1)
  Trunk vsans (up) (1)
  Trunk vsans (isolated) (10)
```

```

Trunk vsans (initializing)  ()
Using Profile id 3 (interface GigabitEthernet4/3)
Peer Information
  Peer Internet address is 43.1.1.1 and port is 3225
  Special Frame is disabled
Maximum number of TCP connections is 2
Time Stamp is disabled
B-port mode disabled
TCP Connection Information
  2 Active TCP connections
    Control connection: Local 43.1.1.2:3225, Remote 43.1.1.1:65532
    Data connection: Local 43.1.1.2:3225, Remote 43.1.1.1:65534
  30 Attempts for active connections, 0 close of connections
TCP Parameters
  Path MTU 1500 bytes
  Current retransmission timeout is 300 ms
  Round trip time: Smoothed 10 ms, Variance: 5
  Advertised window: Current: 122 KB, Maximum: 122 KB, Scale: 1
  Peer receive window: Current: 114 KB, Maximum: 114 KB, Scale: 1
  Congestion window: Current: 2 KB, Slow start threshold: 1048560 KB
5 minutes input rate 64 bits/sec, 8 bytes/sec, 0 frames/sec
5 minutes output rate 64 bits/sec, 8 bytes/sec, 0 frames/sec
808 frames input, 75268 bytes
  808 Class F frames input, 75268 bytes
  0 Class 2/3 frames input, 0 bytes
  0 Error frames timestamp error 0
806 frames output, 74712 bytes
  806 Class F frames output, 74712 bytes
  0 Class 2/3 frames output, 0 bytes
  0 Error frames 0 reass frames

```

### **Example 22-15 Displays Detailed FCIP Interface Counter Information**

```

switch# show interface fcip 3 counters
fcip3
TCP Connection Information
  2 Active TCP connections
    Control connection: Local 43.1.1.2:3225, Remote 43.1.1.1:65532
    Data connection: Local 43.1.1.2:3225, Remote 43.1.1.1:65534
  30 Attempts for active connections, 0 close of connections
TCP Parameters
  Path MTU 1500 bytes
  Current retransmission timeout is 300 ms
  Round trip time: Smoothed 10 ms, Variance: 5
  Advertised window: Current: 122 KB, Maximum: 122 KB, Scale: 1
  Peer receive window: Current: 114 KB, Maximum: 114 KB, Scale: 1
  Congestion window: Current: 2 KB, Slow start threshold: 1048560 KB
5 minutes input rate 64 bits/sec, 8 bytes/sec, 0 frames/sec
5 minutes output rate 64 bits/sec, 8 bytes/sec, 0 frames/sec
814 frames input, 75820 bytes
  814 Class F frames input, 75820 bytes
  0 Class 2/3 frames input, 0 bytes
  0 Error frames timestamp error 0
812 frames output, 75264 bytes
  812 Class F frames output, 75264 bytes
  0 Class 2/3 frames output, 0 bytes
  0 Error frames 0 reass frames

```

### **Example 22-16 Displays Brief FCIP Interface Counter Information**

```

switch# show interface fcip 3 counters brief
-----

```

Interface	Input (rate is 5 min avg)		Output (rate is 5 min avg)	
	Rate Mbits/s	Total Frames	Rate Mbits/s	Total Frames
fcip3	9	0	9	0

### Example 22-17 Displays the FCIP Interface Description

```
switch# show interface fcip 51 description
FCIP51
    Sample FCIP interface
```

### Example 22-18 Displays FCIP Profiles

```
switch# show fcip profile
```

ProfileId	Ipaddr	TcpPort
1	10.10.100.150	3225
2	10.10.100.150	3226
40	40.1.1.2	3225
100	100.1.1.2	3225
200	200.1.1.2	3225

### Example 22-19 Displays the Specified FCIP Profile Information

```
switch# show fcip profile 7
FCIP Profile 7
    Internet Address is 47.1.1.2 (interface GigabitEthernet4/7)
    Listen Port is 3225
    TCP parameters
        SACK is disabled
        PMTU discovery is enabled, reset timeout is 3600 sec
        Keep alive is 60 sec
        Minimum retransmission timeout is 300 ms
        Maximum number of re-transmissions is 4
        Send buffer size is 0 KB
        Maximum allowed bandwidth is 1000000 kbps
        Minimum available bandwidth is 15000 kbps
        Estimated round trip time is 1000 usec
```

[Example 22-20](#) provides a sample output of FCIP counters when the write acceleration feature is enabled.

### Example 22-20 Displays IP Compression Counters in a FCIP Interface

```
switch# show interface fcip 8 counters
fcip8
    TCP Connection Information
        2 Active TCP connections
            Control connection: Local 10.2.2.1:3225, Remote 10.2.2.2:65439
            Data connection: Local 10.2.2.1:3225, Remote 10.2.2.2:65441
        4 Attempts for active connections, 0 close of connections
    TCP Parameters
        Path MTU 1500 bytes
        Current retransmission timeout is 200 ms
        Round trip time: Smoothed 2 ms, Variance: 1
        Advertized window: Current: 14 KB, Maximum: 14 KB, Scale: 9
        Peer receive window: Current: 14 KB, Maximum: 17 KB, Scale: 9
```



```

Congestion window: Current: 10 KB, Slow start threshold: 112 KB
5 minutes input rate 760 bits/sec, 95 bytes/sec, 0 frames/sec
5 minutes output rate 912 bits/sec, 114 bytes/sec, 0 frames/sec
9379771 frames input, 16906568212 bytes
  638 Class F frames input, 58752 bytes
  9379133 Class 2/3 frames input, 16906509460 bytes
  7908669 Reass frames
  0 Error frames timestamp error 0
9229787 frames output, 16569073984 bytes
  638 Class F frames output, 60128 bytes
  9229149 Class 2/3 frames output, 16569013856 bytes
  0 Error frames
Write Accelerator statistics
18609558 packets in      10219163 packets out
0 frames dropped 0 CRC errors
0 rejected due to table full
0 ABTS sent      6 ABTS received
0 tunnel synchronization errors
485136 writes recd   485136 XFER_RDY sent (host)
485136 XFER_RDY rcvd (host)
0 XFER_RDY not proxied due to flow control (host)
0 bytes queued for sending
0 estimated bytes queued on the other side for sending
0 times TCP flow ctrl (target)
0 bytes current TCP flow ctrl (target)
IP compression statistics
10044 rxbytes          0 rxbytes compressed
10044 txbytes          6460 txbytes compressed

```

## FCIP High Availability

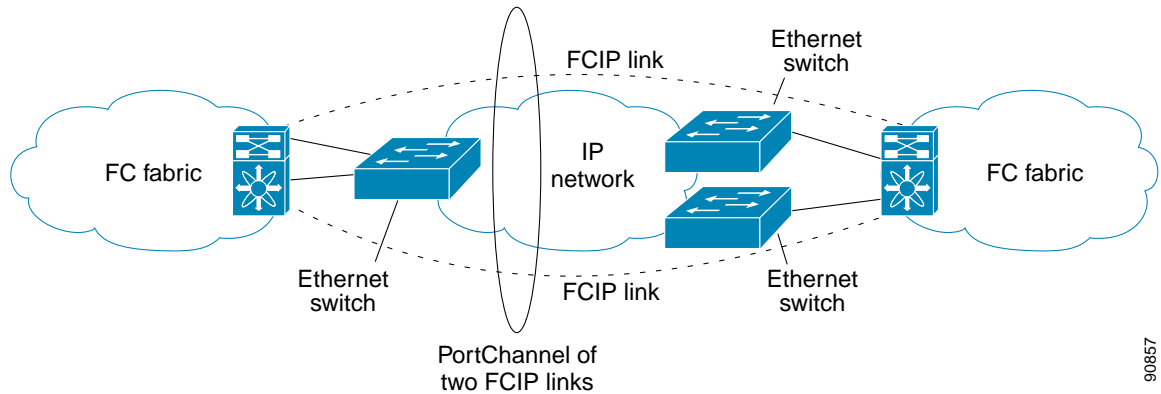
The following high availability solutions are available for FCIP configurations:

- [Fibre Channel PortChannels, page 22-42](#)
- [FSPF, page 22-42](#)
- [VRRP, page 22-43](#)
- [Ethernet PortChannels, page 22-43](#)

## Fibre Channel PortChannels

Figure 22-14 provides an example of a PortChannel-based load balancing configuration. To perform this configuration, you need two IP addresses on each SAN island. This solution addresses link failures.

Figure 22-14 PortChannel Based Load Balancing



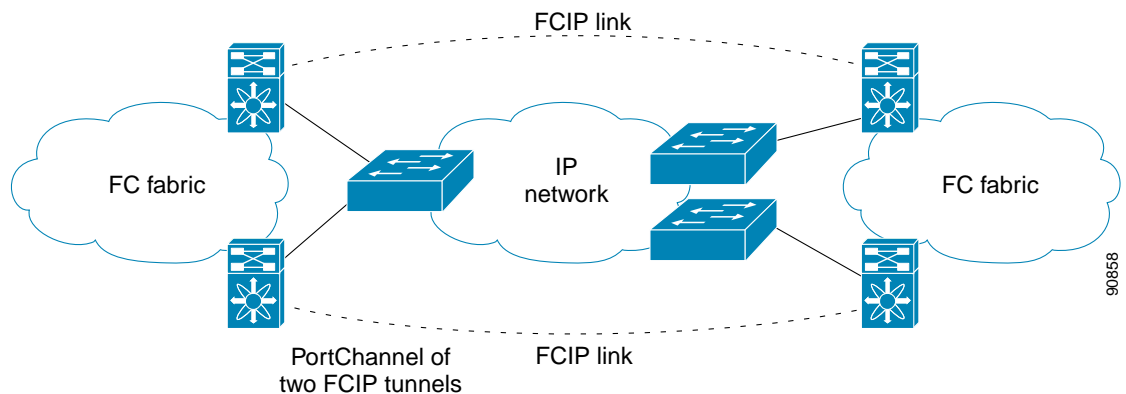
The following characteristics set Fibre Channel PortChannel solutions apart from other solutions:

- The entire bundle is one logical (E)ISL link.
- All FCIP links in the PortChannel should be across the same two switches.
- The Fibre Channel traffic is load balanced across the FCIP links in the PortChannel.

## FSPF

Figure 22-15 displays a FSPF-based load balancing configuration example. This configuration requires two IP addresses on each SAN island, and addresses IP and FCIP link failures.

Figure 22-15 FSPF-Based Load Balancing



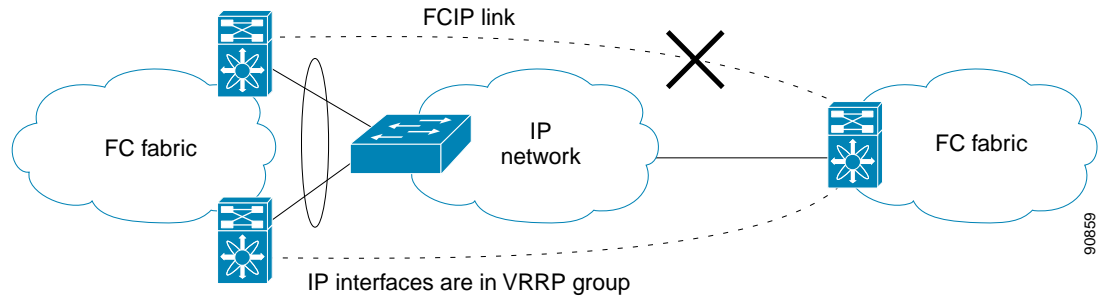
The following characteristics set FSPF solutions apart from other solutions:

- Each FCIP link is a separate (E)ISL.
- The FCIP links can connect to different switches across two SAN islands.
- The Fibre Channel traffic is load balanced across the FCIP link.

## VRRP

Figure 22-16 displays a VRRP-based high availability FCIP configuration example. This configuration requires at least two physical Gigabit Ethernet ports connected to the Ethernet switch on the island where you need to implement high availability using VRRP.

Figure 22-16 VRRP-Based High Availability



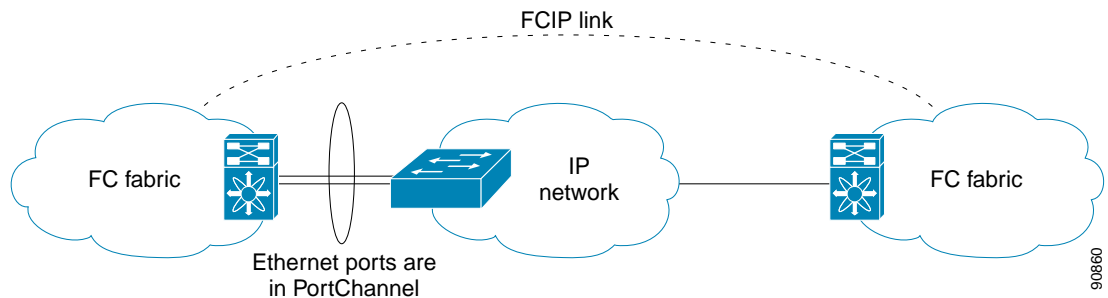
The following characteristics set VRRP solutions apart from other solutions:

- If the active VRRP port fails, the standby VRRP port takes over the VRRP IP address.
- When the VRRP switchover happens, the FCIP link automatically disconnects and reconnects.
- This configuration has only one FCIP (E)ISL link.

## Ethernet PortChannels

Figure 22-17 displays an Ethernet PortChannel-based high availability FCIP example. This solution addresses the problem caused by individual Gigabit Ethernet link failures.

Figure 22-17 Ethernet PortChannel-Based High Availability



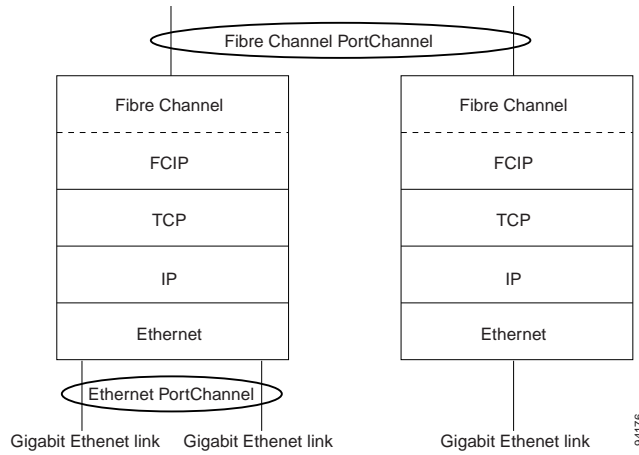
The following characteristics set Ethernet PortChannel solutions apart from other solutions:

- The Gigabit Ethernet link level redundancy ensures a transparent failover if one of the Gigabit Ethernet links fails.
- Two Gigabit Ethernet ports in one Ethernet PortChannel appear like one logical Gigabit Ethernet link.
- The FCIP link stays up during the failover.

## Ethernet PortChannels and Fibre Channel PortChannels

Ethernet PortChannels offer Ethernet-level redundancy and Fibre Channel PortChannels offer (E)ISL-level redundancy. FCIP is unaware of any Ethernet PortChannels or Fibre Channel PortChannels. Fibre Channel PortChannels are unaware of any Ethernet PortChannels, and there is no mapping between the two (see [Figure 22-18](#)).

**Figure 22-18 PortChannels at the Fibre Channel and Ethernet Levels**



To configure Fibre Channel PortChannels, see [Chapter 12, “Configuring PortChannels.”](#) To configure Ethernet PortChannels, see the [“Ethernet PortChannel Aggregation”](#) section on page 22-15.

# Configuring iSCSI

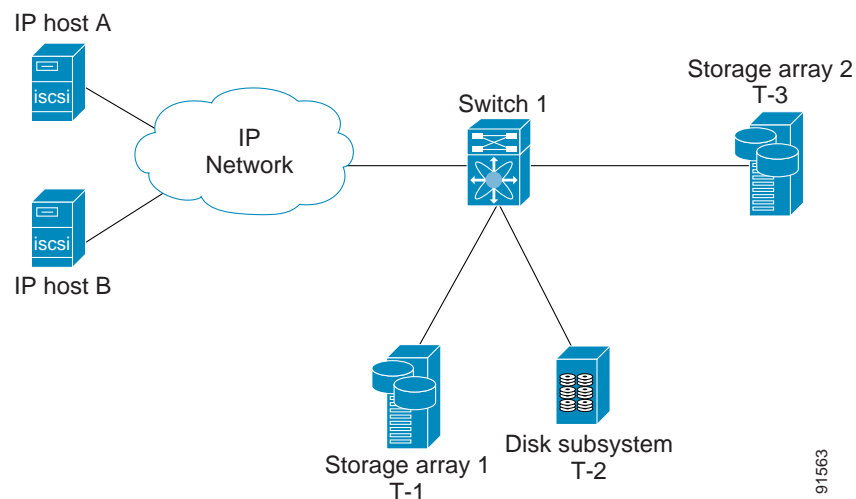
This section includes the following topics:

- [About iSCSI, page 22-45](#)
- [Enabling iSCSI, page 22-47](#)
- [Presenting Fibre Channel Targets as iSCSI Targets, page 22-48](#)
- [iSCSI Virtual Target Configuration Examples, page 22-53](#)
- [Presenting iSCSI Hosts as Virtual Fibre Channel Hosts, page 22-55](#)
- [Configuring iSCSI Proxy Initiators, page 22-59](#)
- [Access Control in iSCSI, page 22-61](#)
- [iSCSI User Authentication, page 22-64](#)
- [Assigning VSAN Membership to iSCSI Hosts, page 22-57](#)
- [Advanced iSCSI Configuration, page 22-66](#)
- [Displaying iSCSI Information, page 22-68](#)
- [iSCSI High Availability, page 22-82](#)
- [iSCSI Setup Guidelines and Scenarios, page 22-85](#)

## About iSCSI

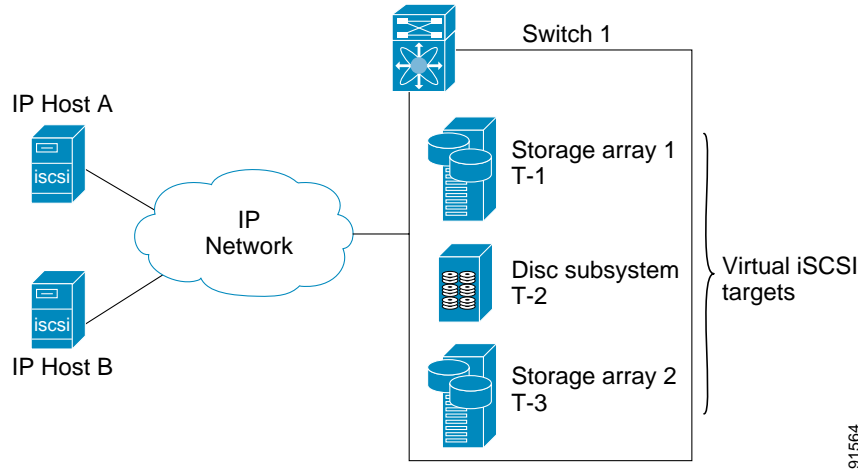
The IPS module provides transparent SCSI routing by default. IP hosts using the iSCSI protocol can transparently access targets on the Fibre Channel network. [Figure 22-19](#) provides an example of a typical configuration of iSCSI hosts with access to a Fibre Channel SAN.

**Figure 22-19 Typical IP to Fibre Channel SAN Configuration**



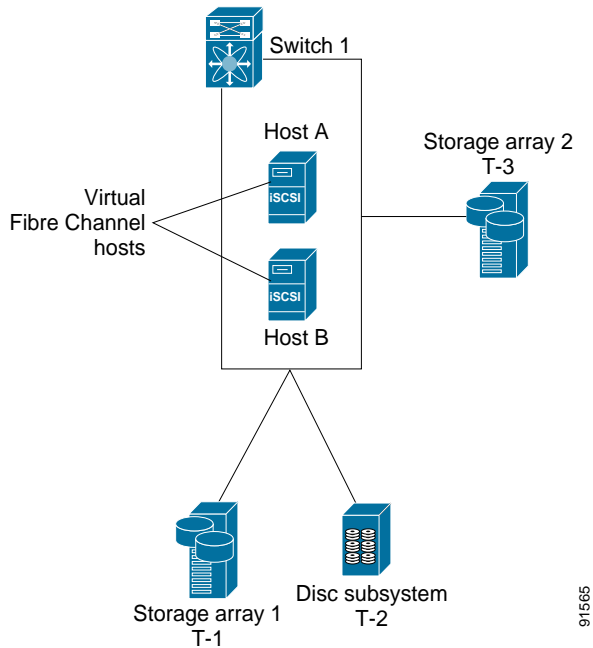
IPS modules enable you to create virtual iSCSI targets and then maps them to physical Fibre Channel targets available in the Fibre Channel SAN. They presents the Fibre Channel targets to IP hosts as if the physical targets were attached to the IP network (see Figure 22-20).

Figure 22-20 iSCSI View



In conjunction with presenting Fibre Channel targets to iSCSI hosts, the IPS module presents each iSCSI host as a Fibre Channel host (in transparent mode), that is, a host bus adapter (HBA) to the Fibre Channel storage device. The storage device responds to each IP host as if it were a Fibre Channel host connected to the Fibre Channel network (see Figure 22-21).

Figure 22-21 Fibre Channel SAN View



Note

Refer to the IETF standards for IP storage at <http://www.ietf.org> for information on the iSCSI protocol.

## Enabling iSCSI

To begin configuring the iSCSI feature, you must explicitly enable iSCSI on the required switches in the fabric. By default, this feature is disabled in all switches in the Cisco MDS 9000 Family.

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.

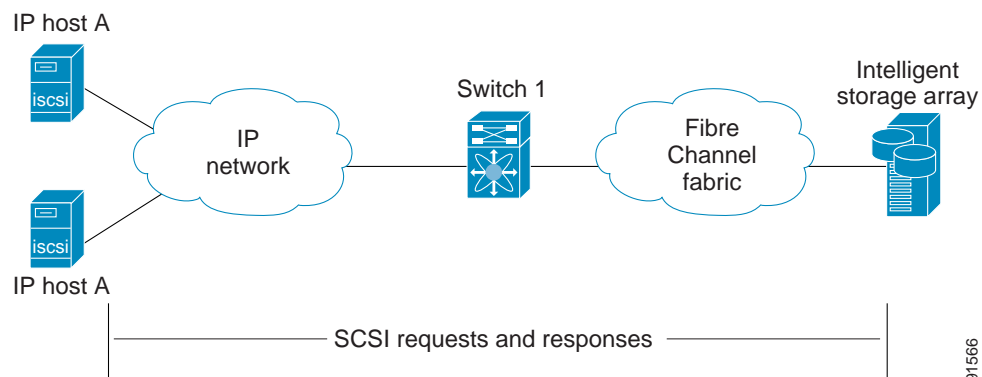
To enable iSCSI on any participating switch, follow these steps:

	Command	Purpose
Step 1	switch# <code>confi g t</code>	Enters configuration mode.
Step 2	switch(config)# <code>iscsi enable</code>	Enables iSCSI on that switch.
	switch(config)# <code>no iscsi enable</code>	Disables (default) iSCSI on that switch.

## Routing iSCSI Requests and Responses

The iSCSI feature consists of routing iSCSI requests and responses between hosts in an IP network and Fibre Channel storage devices in the Fibre Channel SAN that are accessible from any Fibre Channel interface of the Cisco MDS 9000 Family switch (see [Figure 22-22](#)).

**Figure 22-22 Routing iSCSI Requests and Responses for Transparent iSCSI Routing**

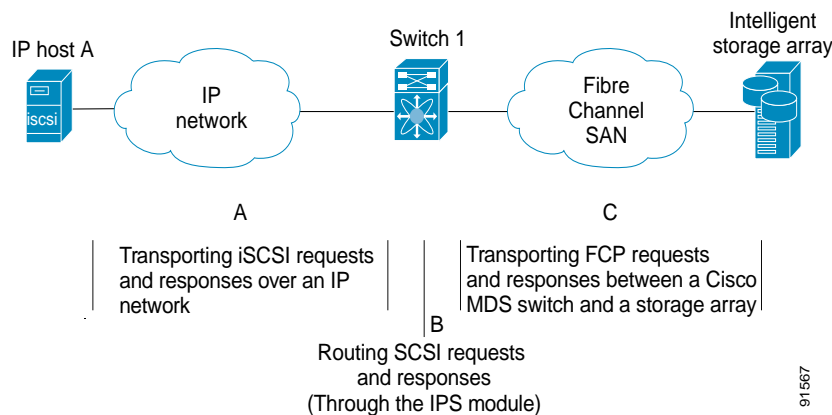


Each iSCSI host that requires access to storage through the IPS module needs to have a compatible iSCSI driver installed. (The Cisco.com website at <http://www.cisco.com/cgi-bin/tablebuild.pl/sn5420-scsi> provides a list of compatible drivers). Using the iSCSI protocol, the iSCSI driver allows an iSCSI host to transport SCSI requests and responses over an IP network. From the host operating system perspective, the iSCSI driver appears to be a SCSI transport driver similar to a Fibre Channel driver for a peripheral channel in the host. From the storage device perspective, each IP host appears as a Fibre Channel host.

Routing SCSI from the IP host to the Fibre Channel storage device consists of the following main actions (see [Figure 22-23](#)):

- The iSCSI requests and responses are transported over an IP network between the hosts and the IPS module.
- The SCSI requests and responses are routed between the hosts on an IP network and the Fibre Channel storage device (converting iSCSI to FCP and vice versa). The IPS module performs this routing.
- The FCP requests or responses are transported between the IPS module and the Fibre Channel storage devices.

**Figure 22-23 Transparent SCSI Routing Actions**



Note

FCP (the Fibre Channel equivalent of iSCSI) carries SCSI commands over a Fibre Channel SAN.

## Presenting Fibre Channel Targets as iSCSI Targets

The IPS module presents physical Fibre Channel targets as iSCSI targets allowing them to be accessed by iSCSI hosts. It does this in one of two ways:

- Dynamic importing—use if all logical units (LUs) in all Fibre Channel storage targets are made available to iSCSI hosts (subject to VSAN and zoning).
- Static importing—use if iSCSI hosts are restricted to subsets of LUs in the Fibre Channel targets and additional iSCSI access control is needed (see the [“Access Control in iSCSI”](#) section on [page 22-61](#)). Also, static importing allows automatic failover if the LUs of the Fibre Channel targets are reached by redundant Fibre Channel ports (see the [“High Availability Static Target Importing”](#) section on [page 22-51](#)).



Note

The IPS module does not import Fibre Channel targets to iSCSI by default. Either dynamic or static mapping must be configured before the IPS module makes Fibre Channel targets available to iSCSI initiators. When static mapping is configured, statically mapped Fibre Channel targets have a configured name. When dynamic mapping is configured, the dynamically-mapped Fibre Channel targets have the name created by the conventions explained in this section.



## Dynamic Importing

The IPS module maps each physical Fibre Channel target port as one iSCSI target. That is, all LU accessible through the physical storage target port are available as iSCSI LUs with the same LU number (LUN) as in the storage target.

For example, if an iSCSI target was created for a Fibre Channel target port with pWWN 31:00:11:22:33:44:55:66 and that pWWN contains LUN 0 through 2, those LUNs would become available to an IP host as LUNs 0 through 2 as well.



### Note

If you have configured a switch name, then the switch name is used instead of the management IP address. If you have not configured a switch name, the management IP address is used.

The iSCSI target node name is created automatically using the iSCSI qualified name (IQN) format. The iSCSI qualified name is restricted to a maximum name length of 223 alphanumeric characters and a minimum length of 16 characters.

The IPS module creates an IQN formatted iSCSI node name using the following conventions:

- IPS ports that are not part of a VRRP group use this format:

```
iqn.1987-05.com.cisco:05.<mgmt-ip-address>.<slot#>-<port#>-<sub-intf#>.<Target-pWWN>
```

- IPS ports that are part of a VRRP group use this format:

```
iqn.1987-05.com.cisco:05.vrrp-<vrrp-ID#>-<vrrp-IP-addr>.<Target-pWWN>
```

- Ports that are part of a PortChannel use this format:

```
iqn.1987-02.com.cisco:05.<mgmt-ip-address>.pc-<port-ch-sub-intf#>.<Target-pWWN>
```



### Note

With this format, each IPS port in a Cisco MDS 9000 Family switch creates a different iSCSI target node name for the same Fibre Channel target.

Use the **iscsi import target fc** command to enable dynamic importing of Fibre Channel targets into iSCSI.

To dynamically import Fibre Channel targets, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi import target fc</b>	IPS modules dynamically import each Fibre Channel target in the Fibre Channel SAN to the IP network. The automatically created iSCSI target node names use the IQN format.  <b>Note</b> Each iSCSI initiator may not have access to all targets depending on the configured access control mechanisms.

## Static Importing

You can manually (statically) create an iSCSI target and assign a node name to it. A statically mapped iSCSI target can either contain the whole FC target port, or it can contain one or more LUs from a Fibre Channel target port.

To create a static iSCSI virtual target for the entire Fibre Channel target port, follow these steps:

	Command	Purpose
Step 1	<code>switch# config terminal</code> <code>switch(config)#</code>	Enters configuration mode.
Step 2	<code>switch(config)# iscsi virtual-target name</code> <code>iqn.1987-02.com.cisco.initiator</code> <code>switch(config- (iscsi-tgt))#</code>	Creates the iSCSI target name <code>iqn.1987-02.com.cisco.initiator</code> .
Step 3	<code>switch(config- (iscsi-tgt))# pwwn</code> <code>26:00:01:02:03:04:05:06</code>	Maps a virtual target node to a Fibre Channel target. One iSCSI target cannot contain more than one Fibre Channel target.  Do not specify the LUN if you wish to map the whole Fibre Channel target to an iSCSI target. All Fibre Channel target LUNs are exposed to iSCSI.  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 cannot use this option.
	<code>switch(config- (iscsi-tgt))# pwwn</code> <code>26:00:00:00:00:11:00:11 fc-lun 1 iscsi-lun 1</code>	Maps a virtual target using LUN mapping options.

See the “[iSCSI-Based Access Control](#)” section on page 22-62 for more information on controlling access to statically-imported targets.

For multiple interfaces configured with iSNS (see the “[Configuring Storage Name Services](#)” section on page 22-98), a different static virtual target name has to be created for each interface tagged to an iSNS profile and each static virtual target must be advertised only from one interface (see the “[Advertising iSCSI Targets](#)” section on page 22-50)

## Advertising iSCSI Targets

You can limit the Gigabit Ethernet interfaces over which static iSCSI targets are advertised. By default iSCSI targets are advertised on all Gigabit Ethernet interfaces, subinterfaces, PortChannel interfaces, and PortChannel subinterfaces.

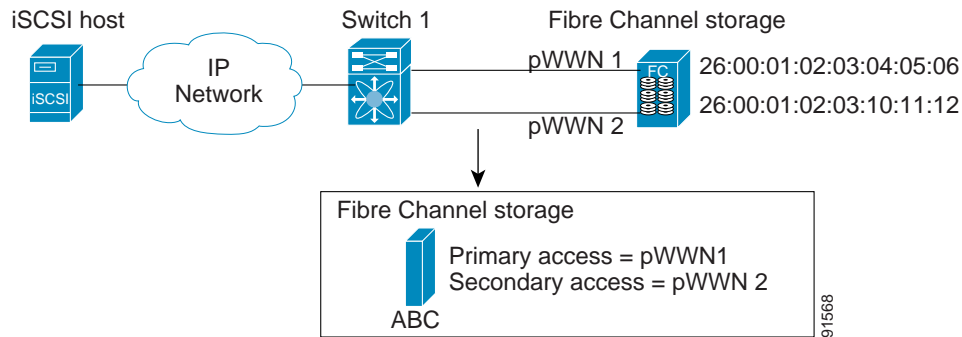
To configure a specific interface that should be advertised by a static iSCSI virtual target, follow these steps:

	Command	Purpose
Step 1	<code>switch(config- (iscsi-tgt))# advertise</code> <code>interface GigabitEthernet 2/5</code>	Advertises the virtual target only on the specified interface. By default, it is advertised on all interfaces in all IPS modules.
	<code>switch(config- (iscsi-tgt))# no</code> <code>advertise interface GigabitEthernet 2/5</code>	Removes this interfaces from the list of interfaces from which this target is advertised.

## High Availability Static Target Importing

Statically imported iSCSI targets have an additional option to provide a secondary pWWN for the Fibre Channel target. This can be used when the physical Fibre Channel target is configured to have an LU visible across redundant ports. When the active port fails, the secondary port becomes active and the iSCSI session switches to use the new active port (see [Figure 22-24](#)).

**Figure 22-24 Static Target Importing Through Two Fibre Channel Ports**



In [Figure 22-24](#), you can create a virtual iSCSI target that is mapped to both pWWN1 and pWWN2 to provide redundant access to the Fibre Channel targets.

The failover to secondary port is done transparently by the IPS port without impacting the iSCSI session from the host. All outstanding I/O are terminated with a check condition status when the primary port fails. New I/O received while the failover has not completed will receive a busy status.



### Tip

If you use LUN mapping, you can define a different secondary Fibre Channel LUN if the LU number is different.

Enable the optional **revert-primary-port** option to direct the IPS port to switch back to the primary port when the primary port is up again. If this option is disabled (default) and the primary port is up again after a switchover, the old sessions will remain with the secondary port and does not switch back to the primary port. However, any new session will use the primary port. This is the only situation when both the primary and secondary ports are used at the same time.

To create a static iSCSI virtual target, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi virtual-target name</b> <b>iqn.1987-02.com.cisco.initiator</b>	Creates the iSCSI target name iqn.1987-02.com.cisco.initiator.
Step 3	switch(config-(iscsi-tgt))# <b>pwwn 26:00:01:02:03:04:05:06</b> <b>secondary-pwwn 26:00:01:02:03:10:11:12</b>	Configures the primary and secondary ports for this virtual target.

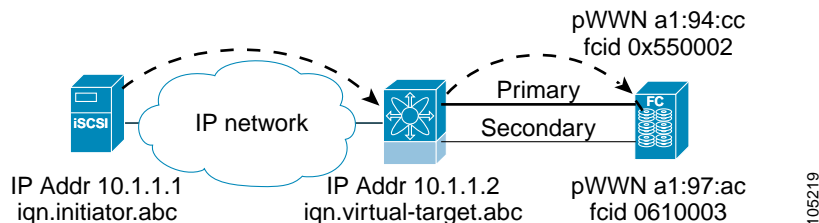
	Command	Purpose
Step 4	<code>switch(config-(iscsi-tgt))# revert-primary-port</code>	Configures the session failover redundancy for this virtual-target to switch all sessions back to primary port when the primary port comes back up.
	<code>switch(config-(iscsi-tgt))# no revert-primary-port</code>	Directs the switch to continue using the secondary port for existing sessions and to use the primary port for new sessions (default)
	<code>switch(config-(iscsi-tgt))# pwwn 26:00:01:02:03:04:05:06 secondary-pwwn 26:00:01:02:03:10:11:12 fc-lun 1 iscsi-lun 0 sec-lun 3</code>	Configures the primary and secondary ports for this virtual target with  lun mapping and different LU number on the secondary FC port.

### Storage Port Failover LUN Trespass

In addition to the high availability of statically imported iSCSI targets, the trespass feature is available (as of Cisco MDS SAN-OS Release 1.3(x)) to enable the export of LUs, on an active port failure, from the active to the passive port of a statically imported iSCSI target.

In physical Fibre Channel targets, which are configured to have LUs visible over two Fibre Channel N-ports, when the active port fails, the passive port takes over. Some physical Fibre Channel targets require that the **trespass** command be issued to export the LUs from the active port to the passive port. A statically imported iSCSI target's secondary pWWN option and an additional option of enabling the trespass feature is available for a physical Fibre Channel target with redundant ports. When the active port fails, the passive port becomes active, and if the trespass feature is enabled, the Cisco MDS switch issues a **trespass** command to the target to export the LUs on the new active port. The iSCSI session switches to use the new active port and the exported LUs are accessed over the new active port (see Figure 22-25).

Figure 22-25 Virtual Target with an Active Primary Port



To enable the trespass feature for a static iSCSI virtual target, follow these steps:

	Command	Purpose
Step 1	<code>switch# config terminal switch(config)#</code>	Enters configuration mode.
Step 2	<code>switch(config)# iscsi virtual-target name 1987-02.com.cisco.initiator switch(config-(iscsi-tgt))#</code>	Creates the iSCSI target name iqn.1987-02.com.cisco.initiator.

	Command	Purpose
Step 3	switch(config-(iscsi-tgt))# <b>pwwn</b> 50:00:00:a1:94:cc <b>secondary-pwwn</b> 50:00:00:a1:97:ac	Maps a virtual target node to a Fibre Channel target and configures a secondary pWWN.
Step 4	switch(config-(iscsi-tgt))# <b>trespass</b>	Enables the trespass feature.
	switch(config-(iscsi-tgt))# <b>no trespass</b>	Disables the trespass feature (default).

Use the **show iscsi virtual-target** command to verify.

```
switch# show iscsi virtual-target iqn.1987-02.com.cisco.initiator
target: 1987-02.com.cisco.initiator
  Port WWN 00:00:00:00:00:00:00:00
  Configured node
  all initiator permit is disabled
  trespass support is enabled
```

## iSCSI Virtual Target Configuration Examples

This section provides three examples of virtual target configurations.

### Example 1

This example assigns the whole Fibre Channel target as a virtual iSCSI target. All LUNs that are part of the Fibre Channel target are available as part of the iSCSI target (see [Figure 22-26](#)).

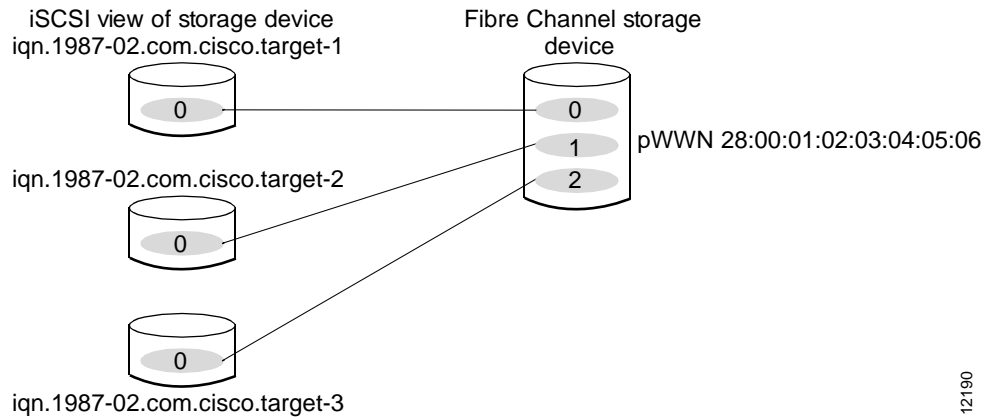
```
iscsi virtual-target name iqn.1987-02.com.cisco.initiator-1
  pWWN 28:00:01:02:03:04:05:06
```

**Figure 22-26 Assigning iSCSI Node Names**

### Example 2

This example maps a subset of LUNs of a Fibre Channel target to three iSCSI virtual targets. Each iSCSI target only has one LUN (see [Figure 22-27](#)).

```
iscsi virtual-target name iqn.1987-02.com.cisco.initiator-1
  pWWN 28:00:01:02:03:04:05:06 fc-lun 0 iscsi-lun 0
iscsi virtual-target name iqn.1987-02.com.cisco.initiator-2
  pWWN 28:00:01:02:03:04:05:06 fc-lun 1 iscsi-lun 0
iscsi virtual-target name iqn.1987-02.com.cisco.initiator-3
  pWWN 28:00:01:02:03:04:05:06 fc-lun 2 iscsi-lun 0
```

**Figure 22-27 Mapping LUNs to a iSCSI Node Name**

112190

**Example 3**

This example maps three subsets of Fibre Channel LUN targets to three iSCSI virtual targets. Two iSCSI targets have one LUN and the third iSCSI target has two LUNs (see [Figure 22-28](#)).

```
iscsi virtual-target name iqn.1987-02.com.cisco.initiator-1
  pWWN 28:00:01:02:03:04:05:06 fc-lun 0 iscsi-lun 0
iscsi virtual-target name iqn.1987-02.com.cisco.initiator-2
  pWWN 28:00:01:02:03:04:05:06 fc-lun 1 iscsi-lun 0
iscsi virtual-target name iqn.1987-02.com.cisco.initiator-3
  pWWN 28:00:01:02:03:04:05:06 fc-lun 2 iscsi-lun 0
  pWWN 28:00:01:02:03:04:05:06 fc-lun 3 iscsi-lun 1
```

**Figure 22-28 Mapping LUNs to Multiple iSCSI Node Names**

## Presenting iSCSI Hosts as Virtual Fibre Channel Hosts

The iSCSI hosts are mapped to virtual Fibre Channel hosts in one of two ways (see [Figure 22-21](#)):

- Dynamic mapping (default)—use if no access control is done on the Fibre Channel target. An iSCSI host may use different pWWNs each time it connects to a Fibre Channel target.
- Static mapping—use if an iSCSI host should always have the same pWWN or nWWN each time it connects to a Fibre Channel target.

### Dynamic Mapping

When an iSCSI host connects to the IPS module using the iSCSI protocol, a virtual N port is created for the host. The nWWNs and pWWNs are dynamically allocated from the switch's Fibre Channel WWN pool. The IPS module registers this N port in the Fibre Channel SAN. The IPS module continues using that nWWN and pWWN to represent this iSCSI host until it no longer has a connection to any iSCSI target through that IP storage port.

At that point, the virtual Fibre Channel host is taken offline from the Fibre Channel SAN and the nWWNs and pWWNs are released back to the switch's Fibre Channel WWN pool. These addresses become available for assignment to other iSCSI hosts requiring access to Fibre Channel SANs. When a dynamically mapped iSCSI initiator has multiple sessions to multiple Fibre Channel targets, each session can use the same pWWN and nWWN as long as it uses the same node name in the iSCSI login message.

If the host has multiple network interfaces (each having different IP addresses), you can treat each IP address as a different iSCSI initiator host by using the **switchport initiator id ip-address** command.

### Identifying Initiators

By default, the switch uses the iSCSI node name to identify the initiator.

An iSCSI initiator is identified in one of two ways:

- By iSCSI node name—An initiator with multiple IP addresses (multiple interface cards—NICs or multiple network interfaces) has one virtual N port, assuming it uses the same iSCSI initiator name from all interfaces.
- By IP address—A virtual N port is created for each IP address it uses to log in to iSCSI targets.

Use the **switchport initiator id name** command to identify the iSCSI initiator using the iSCSI node name and the **switchport initiator id ip-address** command to identify the iSCSI initiator using the IP address,

To identify the initiator using the IP address, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface iscsi 4/1</b> switch(config-if)#	Selects the iSCSI interface on the switch that identifies all the initiators.
Step 3	switch(config-if)# <b>switchport initiator id ip-address</b>	Identifies the iSCSI initiator based on the IP address.
	switch(config-if)# <b>switchport initiator id name</b>	Identifies the iSCSI initiator based on the initiator node name.

## Static Mapping

With dynamic mapping, each time the iSCSI host connects to the IPS module a new Fibre Channel N port is created and the nWWNs and pWWNs allocated for this N port may be different. Use the static mapping method to obtain the same nWWN and pWWNs for the iSCSI host each time it connects to the IPS module.

Static mapping can be used on the IPS module to access intelligent Fibre Channel storage arrays that have access control and LUN mapping or masking configuration based on the initiator's pWWNs and/or nWWNs.



### Note

If an iSCSI host connects to multiple IPS ports, each port independently creates one virtual N port for the host. If static mapping is used, enough pWWNs should be configured for as many IPS ports to which a host connects.

You can implement static mapping in one of two ways:

- **Manual assignment**—You can specify your own unique WWN by providing them during the configuration process.
- **System assignment**—When a static mapping configuration is created, one nWWN and/or one or more pWWNs are allocated from the switch's Fibre Channel WWN pool and the mapping is kept permanent.

This assignment uses the **system-assign** option.



### Tip

We recommend using the **system-assign** option. If you manually assign a WWN, you must ensure its uniqueness (see the [“Configuring World Wide Names”](#) section on page 29-18).

To configure static mapping (using the **name** option) for an iSCSI initiator, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi initiator name</b> <b>iqn.1987-02.com.cisco.initiator</b> switch(config-iscsi-init)#	Configures an iSCSI initiator using the iSCSI name of the initiator node. The maximum name length is restricted to 223 alphanumeric characters. The minimum length is 16.
	switch(config)# <b>no iscsi initiator</b> <b>name iqn.1987-02.com.cisco.initiator</b>	Deletes the configured iSCSI initiator.

To configure static mapping (using the **ip-address** option) for an iSCSI initiator, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi initiator ip</b> <b>address 10.50.0.0</b> switch(config-iscsi-init)#	Configures an iSCSI initiator using the IP address of the initiator node.
	switch(config)# <b>no iscsi initiator ip</b> <b>address 10.50.0.0</b>	Deletes the configured iSCSI initiator.



To assign the WWN for an iSCSI initiator, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-iscsi-init)# static nWWN system-assign</code>	Uses the switch's WWN pool to allocate the nWWN for this iSCSI initiator and keeps it persistent.
	<code>switch(config-iscsi-init)# nWWN 20:00:00:05:30:00:59:11</code>	Assigns the user provided WWN as nWWN for the iSCSI initiator. You can only specify one nWWN for each iSCSI node.
Step 2	<code>switch(config-iscsi-init)# static pWWN system-assign 2</code>	Uses the switch's WWN pool to allocate two pWWNs for this iSCSI initiator and keeps it persistent. The range is from 1 to 64.
	<code>switch(config-iscsi-init)# pWWN 21:00:00:20:37:73:3b:20</code>	Assigns the user provided WWN as pWWN for the iSCSI initiator.



#### Note

If a system-assign option is used to configure WWNs for an iSCSI initiator, when the configuration is backed up to an ASCII file the system-assigned WWNs are also saved. Subsequently if you issue a **write erase** command, you must manually delete the WWN configuration from the ASCII file.

## Making the Dynamic Initiator WWN Mapping Static

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

To permanently keep the automatically assigned nWWN/pWWN mapping, follow these steps:

	Command	Purpose
Step 1	<code>switch# config terminal</code> <code>switch(config)#</code>	Enters configuration mode.
Step 2	<code>switch(config)# iscsi save-initiator name iqn.1987-02.com.cisco.initiator</code>	Saves the nWWN and pWWNs that have automatically been assigned to the iSCSI initiator whose name is specified.
	<code>switch(config)# iscsi save-initiator ip-address 10.10.100.11</code>	Saves the nWWN and pWWNs that have automatically been assigned to the iSCSI initiator whose IP address is specified.
	<code>switch(config)# no iscsi save-initiator name iqn.1987-02.com.cisco.initiator</code>	Removes the static nWWN and pWWNs mapping for the iSCSI initiator whose name is specified.

## Assigning VSAN Membership to iSCSI Hosts

By default, a host is only in VSAN 1 (default VSAN). You can configure an iSCSI host to be a member of one or more VSANs. The IPS module creates one Fibre Channel virtual N port in each VSAN to which the host belongs.

To assign VSAN membership for iSCSI hosts, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi initiator name</b> <b>iqn.1987-02.com.cisco.initiator</b> switch(config-iscsi-init)#	Configures an iSCSI initiator.
Step 3	switch(config-iscsi-init)# <b>vsan 3</b>	Assigns the iSCSI initiator node to a specified VSAN. <b>Note</b> You can assign this host to one or more VSANs.
	switch(config-iscsi-init)# <b>no vsan 5</b>	Removes the iSCSI node from the specified VSAN.

**Note**

When an initiator is configured in any other VSAN (other than VSAN 1), for example VSAN 2, the initiator is automatically removed from VSAN 1. If you also want it to be present in VSAN 1, you must explicitly configure the initiator in VSAN 1.

## Assigning VSANs to a iSCSI Interface

All dynamic iSCSI initiators are members of VSAN 1. The port VSAN of an iSCSI interface is the default VSAN for all dynamic iSCSI initiators. All dynamic iSCSI initiators are members of the port VSAN of the iSCSI interface. The default port VSAN of an iSCSI interface is VSAN 1.

Use the **vsan vsan-number interface iscsi slot/port** command in the VSAN database submode to change the default port VSAN.

**Tip**

This is a 1.3(x) feature. If you downgrade to an earlier release, be sure to delete any assigned VSAN and to issue the **no iscsi interface vsan-membership** command before performing the downgrade procedure.

To change the default port VSAN, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi interface</b> <b>vsan-membership</b>	Enables you to configure VSAN membership for iSCSI interfaces.
Step 3	switch(config)# <b>vsan database</b> switch(config-vsan-db)#	Configures the database for a VSAN. Application specific VSAN parameters cannot be configured from this prompt.
Step 4	switch(config-vsan-db)# <b>vsan 2</b> <b>interface iscsi 2/1</b>	Assigns the membership of the iscsi 2/1 interface to the specified VSAN (VSAN 2).
	switch(config-vsan-db)# <b>no vsan 2</b> <b>interface iscsi 2/1</b>	Reverts to using the default VSAN as the port VSAN of the iSCSI interface.

## Configuring iSCSI Proxy Initiators

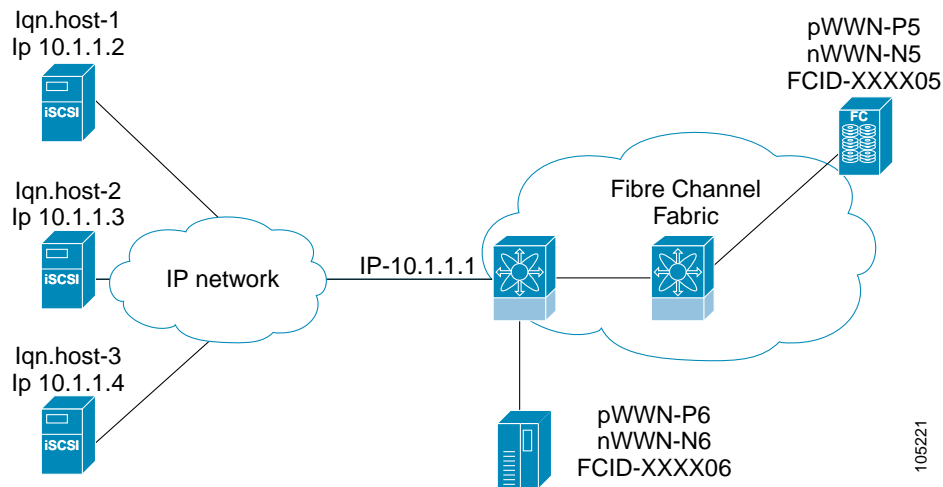


### Note

When an interface is in the proxy initiator mode, you can only configure Fibre Channel access control (zoning) based on the Fibre Channel interface attributes—the WWN pair and available FCIDs. You cannot configure zoning using iSCSI attributes such as the IP address or the iQN name of the iSCSI initiator. To enforce initiator-based access control, use iSCSI based access control (see the “[Access Control in iSCSI](#)” section on page 22-61).

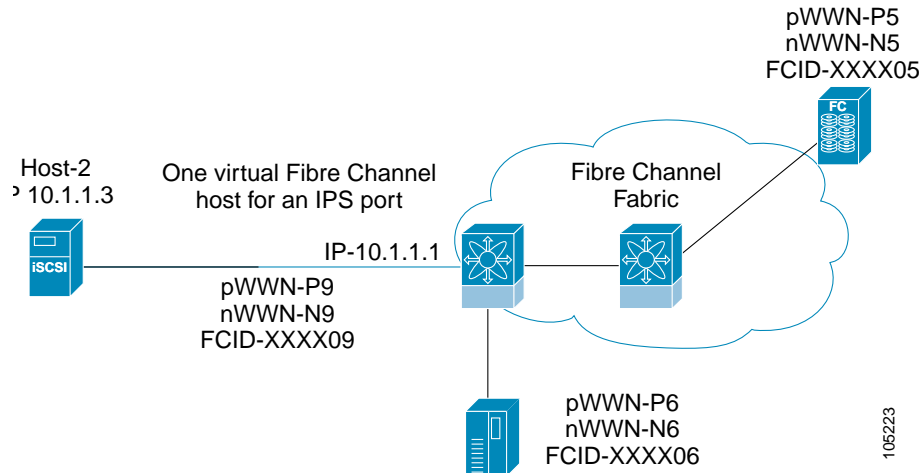
By default, each iSCSI initiator appears as one Fibre Channel initiator in transparent mode in the Fibre Channel fabric. For some storage arrays, this appearance requires the initiator’s pWWN to be manually configured for access control purposes. This process can be quite cumbersome. The proxy initiator feature allows all iSCSI initiators to connect through one IPS port making it appear as one Fibre Channel port per VSAN. It simplifies the task of configuring the pWWN for each new initiator on the storage array, and of configuring Fibre Channel access control such as zoning. This feature along with static target importing (using LUN mapping) results in the configuration being performed only on the switch when a new iSCSI host is added. On the storage array, all LUNs that are used by iSCSI initiators are configured to allow access by the proxy initiator’s pWWN. From the iSCSI perspective, this configuration is no different from the default mode (see [Figure 22-29](#)).

**Figure 22-29 iSCSI View of a Proxy Initiator**



From the Fibre Channel perspective, only one Fibre Channel initiator is visible per VSAN (see [Figure 22-30](#)).

*Figure 22-30 FC View with a Proxy Initiator*



To configure the proxy initiator, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface iscsi 4/1</b> switch(config-if)#	Selects the iSCSI interface on the switch that identifies all the initiators.
Step 3	switch(config-if)# <b>switchport proxy-initiator</b>	Configures the proxy initiator mode using the switch's pWWN/nWWN pool.
	switch(config-if)# <b>no switchport proxy-initiator</b>	Deletes the proxy initiator mode.
Step 4	switch(config-if)# <b>switchport proxy-initiator nWWN 11:11:11:11:11:11:11:11:11:11:11:11:11:11:11:11</b> <b>pwwn 22:22:22:22:22:22:22:22:22:22:22:22:22:22:22:22</b>	Configures the proxy initiator mode using the specified WWNs.
	switch(config-if)# <b>no switchport proxy-initiator nWWN 11:11:11:11:11:11:11:11:11:11:11:11:11:11:11:11</b> <b>pwwn 22:22:22:22:22:22:22:22:22:22:22:22:22:22:22:22</b>	Deletes the proxy initiator mode.

To verify the proxy initiator mode configuration, use the **show interface iscsi** command in EXEC mode (see the “[Displaying Proxy Initiator Information](#)” section on page 22-73).

## iSCSI Initiator Idle Timeout

iSCSI initiator idle timeout specifies the time for which the initiator information is kept after the initiator logs out. The default is 300 seconds.

To configure the initiator idle timeout, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi initiator</b> <b>idle-timeout 10</b>	Configures the iSCSI initiators to have an idle timeout value of 10 seconds.

## Access Control in iSCSI

You can control access to each statically mapped iSCSI target by specifying a list of IPS ports on which it is advertised and specifying a list of iSCSI initiator node names allowed to access it. Fibre Channel zoning-based access control and iSCSI-based access control are the two mechanisms by which access control can be provided for iSCSI. Both methods can be used simultaneously.



### Note

This access control is in addition to the existing Fibre Channel access control. The iSCSI initiator has to be in the same VSAN and zone as the physical Fibre Channel target.

## Fibre Channel Zoning-Based Access Control

Zoning is an access control mechanism within a VSAN. The switch's zoning implementation extends the VSAN and zoning concepts from the Fibre Channel domain to cover the iSCSI domain. This extension includes both iSCSI and Fibre Channel features and provides a uniform, flexible access control across a SAN. There are two Fibre Channel zoning access control mechanisms--static and dynamic.

- **Static**—Statically map the iSCSI host to Fibre Channel virtual N port(s). This creates a permanent nWWNs and pWWNs. Next, configure the assigned pWWN into zones, similar to adding a regular Fibre Channel host's pWWN to a zone.
- **Dynamic**—Add the iSCSI host's initiator node name as a member of a zone. When the IP host's Fibre Channel virtual N port is created and the Fibre Channel address (nWWNs and pWWNs) is assigned, Fibre Channel zoning is enforced.

To register an iSCSI initiator in the zone database, follow these steps:

	Command	Purpose
Step 1	switch# <b>config terminal</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>zone name iSCSIzone vsan 1</b> switch(config-zone)	Creates a zone name for the iSCSI devices in the IPS module to be included.

	Command	Purpose
Step 3	<code>switch(config-zone)# member symbolic-nodename iqn.1987-02.com.cisco.initiator1</code>	Adds the device as specified by the node name.
	<code>switch(config-zone)# no member iqn.1987-02.com.cisco.initiator1</code>	Deletes the specified device.
	<code>switch(config-zone)# member ip-address 10.50.1.1</code>	Adds the device identified by the IP address.
	<code>switch(config-zone)# no member ip-address 10.50.1.1</code>	Deletes the identified device.
	<code>switch(config-zone)# member pwwn 20:00:00:05:30:00:59:11</code>	Adds the device identified by the port WWN.
	<code>switch(config-zone)# no member pwwn 20:00:00:05:30:00:59:11</code>	Deletes the device identified by the port WWN
	<code>switch(config-zone)# member ip-address 10.50.1.1 255.255.0.0</code>	Adds all devices in the specified IP subnet.
	<code>switch(config-zone)# no member ip-address 10.50.1.1 255.255.0.0</code>	Deletes all devices in the specified IP subnet.

## iSCSI-Based Access Control

For static iSCSI targets, you can manually configure a list of iSCSI initiators that are allowed to access the targets. The iSCSI initiator is identified by the iSCSI node name or the IP address of the iSCSI host.

By default, static virtual iSCSI targets are not accessible to any iSCSI host. You must explicitly configure accessibility to allow a virtual iSCSI target to be accessed by all hosts. The initiator access list can contain one or more initiators. Each initiator is identified by one of the following:

- iSCSI node names
- IP addresses
- IP subnets

To configure access control in iSCSI, follow these steps:

	Command	Purpose
Step 1	<code>switch# config terminal switch(config)#</code>	Enters configuration mode.
Step 2	<code>switch(config)# iscsi virtual-target name iqn.1987-02.com.cisco.initiator switch(config-(iscsi-tgt))#</code>	Creates the iSCSI target name iqn.1987-02.com.cisco.initiator.
Step 3	<code>switch(config-(iscsi-tgt))# pwwn 26:00:01:02:03:04:05:06 switch(config-(iscsi-tgt))#</code>	Maps a virtual target node to a Fibre Channel target.

	Command	Purpose
Step 4	<code>switch(config-(iscsi-tgt))# initiator iqn.1987-02.com.cisco.initiator1 permit</code>	Allows the specified iSCSI initiator node to access this virtual target. You can issue this command multiple times to allow multiple initiators.
	<code>switch(config-(iscsi-tgt))# no initiator iqn.1987-02.com.cisco.initiator1 permit</code>	Prevents the specified initiator node from accessing virtual targets.
	<code>switch(config-(iscsi-tgt))# initiator ip address 10.50.1.1 permit</code>	Allows the specified IP address to access this virtual target. You can issue this command multiple times to allow multiple initiators.
	<code>switch(config-(iscsi-tgt))# no initiator ip address 10.50.1.1 permit</code>	Prevents the specified IP address from accessing virtual targets.
	<code>switch(config-(iscsi-tgt))# initiator ip address 10.50.1.1 255.255.255.0 permit</code>	Allows all initiators in this subnetwork to access this virtual target.
	<code>switch(config-(iscsi-tgt))# no initiator ip address 10.50.1.1 255.255.255.0 permit</code>	Prevents all initiators in this subnetwork from accessing virtual targets.
	<code>switch(config-(iscsi-tgt))# all-initiator-permit</code>	Allows all initiator nodes to access this virtual target.
	<code>switch(config-(iscsi-tgt))# no all-initiator-permit</code>	Prevents any initiator from accessing virtual targets (default).

## Enforcing Access Control

IPS modules use both iSCSI node name-based and Fibre Channel zoning-based access control lists to enforce access control during iSCSI discovery and iSCSI session creation.

- iSCSI discovery—When an iSCSI host creates an iSCSI discovery session and queries for all iSCSI targets, the IPS module returns only the list of iSCSI targets this iSCSI host is allowed to access based on the access control policies discussed in the previous section.
- iSCSI session creation—When an IP host initiates an iSCSI session, the IPS module verifies if the specified iSCSI target (in the session login request) is a static mapped target, and if true, verifies if the IP host's iSCSI node name is allowed to access the target. If the IP host does not have access, its login is rejected.

The IPS module then creates a Fibre Channel virtual N port (the N port may already exist) for this IP host and does a Fibre Channel name server query for the FCID of the Fibre Channel target pWWN that is being accessed by the IP host. It uses the IP host virtual N port's pWWN as the requester of the name server query. Thus, the name server does a zone-enforced query for the pWWN and responds to the query.

If the FCID is returned by the name server, then the iSCSI session is accepted. Otherwise, the login request is rejected.



### Note

If you connect to the switch from an AIX or HP-UX host, be sure to enable the persistent FC ID feature in the VSAN that connects these hosts (see the [“Enabling Persistent FC IDs”](#) section on page 24-9).

## iSCSI User Authentication

The IPS module supports the iSCSI authentication mechanism to authenticate iSCSI hosts that request access to storage. By default, IPS modules allow CHAP or None authentication of iSCSI initiators. When iSCSI authentication is enabled, the iSCSI hosts must provide user name and password information each time an iSCSI session is established.



### Note

Only the Challenge Handshake Authentication Protocol (CHAP) authentication method is supported.

You can use RADIUS authentication (see the “[Configuring RADIUS](#)” section on page 16-5) or TACACS+ authentication (see the “[Configuring TACACS+](#)” section on page 16-10). If no authentication is configured, local authentication is used.

The **aaa authentication iscsi** default command enables aaa authentication for the iSCSI host.

To configure AAA authentication for an iSCSI user, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>aaa authentication iscsi default group RadServerGrp</b>	Uses RADIUS servers that are added in the group called RadServerGrp for the iSCSI authentication method.
	switch(config)# <b>aaa authentication iscsi default group TacServerGrp</b>	Uses TACACS+ servers that are added in the group called TacServerGrp for the iSCSI authentication method.
	switch(config)# <b>aaa authentication iscsi default local</b>	Uses the local password database for iSCSI CHAP authentication.
	switch(config)# <b>iscsi authentication none</b>	Specifies no authentication configuration
	switch(config)# <b>iscsi authentication chap-none</b>	Specifies that both CHAP or no authentication is allowed. Use this option to override the global configuration that may have been configured to allow only one option—either CHAP or none—but not both.



## Authentication Mechanism

During an iSCSI login, both the iSCSI initiator and target have the option to authenticate each other. By default, the IPS module allows either CHAP authentication or no authentication from iSCSI hosts.



**Note** The authentication for a Gigabit Ethernet interface or subinterface configuration overrides the authentication for the global interface configuration.

If CHAP authentication should always be used, issue the **iscsi authentication chap** command at either the global level or at a per-interface level. If authentication should not be used, issue the **iscsi authentication none** command.

To configure the authentication mechanism for iSCSI, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi authentication chap</b>	Configures CHAP as the default authentication mechanism globally for the Cisco MDS switch. CHAP authentication is required for all iSCSI sessions. The validation is done using RADIUS or local authentication.

To configure the authentication policy for iSCSI sessions to a particular interface, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface GigabitEthernet 2/1.100</b> switch(config-if)#	Selects the Gigabit Ethernet interface.
Step 3	switch(config-if)# <b>iscsi authentication none</b>	Specifies that no authentication is required for iSCSI sessions to the selected interface.

The IPS module verifies the iSCSI host authentication using the local password database, TACACS+, or RADIUS (see the “[Configuring CLI User Profiles](#)” section on page 16-22). If local authentication is used, the **username iscsi-user password iscsi** command assigns a password and a user name for a new user. If the user name does not exist it is created.



**Note** The **iscsi** keyword is mandatory to identify iSCSI users.

To configure iSCSI users for local authentication, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>username iscsiuser password ffsffsfsffs345353554535 iscsi</b>	Configures a user name (iscsiuser) and password (ffsffsfsffs345353554535) in the local database for iSCSI login authentication.

You can restrict an initiator to use a specified CHAP username to connect to a Cisco MDS switch. This configuration is in addition to configuring the iSCSI username and password,

To restrict an initiator to use a specified CHAP username, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>iscsi initiator name iqn.1987-02.com.cisco.init</b> switch(config-iscsi-init)#	Enters the configuration submode for the initiator iqn.1987-02.com.cisco.init.
Step 3	switch(config-iscsi-init)# <b>username user1</b>	Restricts the initiator <code>iqn.1987-02.com.cisco.init</code> to only authenticate using <code>user1</code> as its CHAP username.  <b>Tip</b> Be sure to define <code>user1</code> as an iSCSI user.

## Advanced iSCSI Configuration

Advanced configuration options are available for iSCSI interfaces on a per-IPS port basis. These configurations are similar to the advanced FCIP configurations and are already explained in that section (see the [Advanced FCIP Profile Configuration, page 22-24](#)).

To access these commands from the iSCSI interface, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface iscsi 4/1</b> switch(config-if)#	Selects the iSCSI interface on the switch that identifies all the initiators.
Step 3	switch(config-if)# <b>tcp ?</b> keepalive-timeout max-bandwidth-kbps max-bandwidth-mbps max-retransmissions min-retransmit-time pmtu-enable qos sack-enable send-buffer-size	Provides the TCP parameters available on a per-IPS port basis for iSCSI interfaces.

Cisco MDS switches support the following advanced features for iSCSI interfaces:

- iSCSI listener port—You can configure the TCP port number for the iSCSI interface which listens for new TCP connections. The default port number is 3260. Following that, the iSCSI port only accepts TCP connections on the newly configured port  
See the “[Configuring TCP Listener Ports](#)” section on page 22-25)
- TCP tuning parameters—You can configure the following TCP parameters.
  - The minimum retransmit timeout, keepalive timeout, maximum retransmissions, path MTU, SACK (SACK is enabled by default for iSCSI TCP configurations), window management (The iSCSI defaults are max-bandwidth = 1G, min-available-bandwidth = 70 Mbps, and round-trip-time = 1 ms.), buffer size (default send buffer size for iSCSI is 4096 KB), window congestion (enabled by default and the default burst size is 50 KB.), and maximum delay jitter (enabled by default and the default time is 500 microseconds.).

See the “Minimum Retransmit Timeout” section on page 22-25, “Keepalive Timeout” section on page 22-25, “Maximum Retransmissions” section on page 22-26, “Path MTU” section on page 22-26, “Monitoring Window Congestion” section on page 22-29 and “Estimating Maximum Delay Jitter” section on page 22-29.

- QoS—QoS configurations differ for iSCSI and FCIP interfaces. To set the QoS values, follow these steps:

	Command	Purpose
Step 1	<code>switch(config-if)# tcp qos 3</code>	Configures the control TCP connection. The DSCP value of 3 is applied to all IP packets in this iSCSI interface. The valid range for the iSCSI DSCP value is from 0 to 63.
Step 2	<code>switch(config-profile)# no tcp qos 5</code>	Reverts the switch to its factory default (marks all packets with DCSP value 0).

- Identification of dynamic iSCSI initiator—iSCSI initiators are identified based on their IQN name or their IP address. In the absence of any configuration for the initiator (WWN or VSAN membership), the identifier key is the default connection. By default, the key is the IQN name but can be changed to IP address by toggling this mode.
- Proxy or transparent Initiator—For each iSCSI initiator with iSCSI target sessions, the switch creates a virtual FC initiator with a distinct pair of WWNs per VSAN. For targets that have access control per LUN, the WWN pair of each FC initiator must be configured in the target. The proxy initiator mode can be enabled to facilitate this configuration, in this case, all iSCSI initiators that connect to this iSCSI interface inherit the same WWN pair and create only one virtual FC initiator in each VSAN.

## iSCSI Forwarding Mode

The iSCSI gateway on the IPS module has two modes of forwarding operation:

- The **pass-thru** mode (default): The IPS port converts an iSCSI PDU into an FCP frame or vice versa and then forwards it one frame or PDU at a time. The absence of buffering PDUs or frames keeps the operation latency low. To operate in this mode, the IPS port has to negotiate with its peers a suitable maximum size of the data payload in each frame/PDU. This is done during iSCSI login and FC PLOGI and the value is restricted by the TCP connection's maximum segment size (MSS) and the maximum Fibre Channel data payload size specified by the FC target. This usually results in a smaller maximum payload size than most hosts expect, thus comes the second mode of forwarding.
- The **store-and-forward** mode: The iSCSI client sends and receives an iSCSI data payload at the size it desires. This sometimes results in better performance for the client. The IPS port stores each TCP segment it receives until one full iSCSI PDU is received before converting and forwarding it as Fibre Channel frames to the FC target. In the opposite direction, the IPS port assembles all FC data frames of an exchange to build one iSCSI data-in PDU before forwarding it to the iSCSI client. The limitation on this mode is that the iSCSI CRC data digest cannot be used.



Tip

The header and data digest feature is automatically enabled when iSCSI initiators send requests. This feature cannot be configured and is not available when using the **store-and-forward** mode.

## Displaying iSCSI Information

This section includes the following topics:

- [Displaying iSCSI Interfaces, page 22-68](#)
- [Displaying Global iSCSI Information, page 22-74](#)
- [Displaying iSCSI Sessions, page 22-75](#)
- [Displaying iSCSI Initiators, page 22-76](#)
- [Displaying iSCSI Virtual Targets, page 22-80](#)
- [Displaying IPS Statistics, page 22-80](#)
- [Displaying iSCSI User Information, page 22-82](#)

## Displaying iSCSI Interfaces

Use the **show iscsi interface** command to view the summary, counter, description, and status of the iSCSI interface. Use the output to verify the administrative mode, the interface status, TCP parameters currently used, and brief statistics. See [Example 22-21](#).

### *Example 22-21 Displays the iSCSI Interface Information*

```
switch# show interface iscsi 2/1
iscsi2/1 is up -----> Interface is up
  Hardware is GigabitEthernet
  Port WWN is 20:41:00:05:30:00:50:de
  Admin port mode is ISCSI
  Port mode is ISCSI
  Speed is 1 Gbps
  iSCSI initiator is identified by name
  Number of iSCSI session: 7, Number of TCP connection: 7
  Configured TCP parameters
    Local Port is 3260
    PMTU discover is disabled
    Keepalive-timeout is 1 sec
    Minimum-retransmit-time is 300 ms
    Max-retransmissions 8
    Sack is disabled
    Minimum available bandwidth is 0 kbps
    Estimated round trip time is 0 usec
  5 minutes input rate 265184 bits/sec, 33148 bytes/sec, 690 frames/sec
  5 minutes output rate 375002168 bits/sec, 46875271 bytes/sec, 33833 frames/sec
  iSCSI statistics
    6202235 packets input, 299732864 bytes
      Command 6189718 pdus, Data-out 1937 pdus, 1983488 bytes, 0 fragments
    146738794 packets output, 196613551108 bytes
      Response 6184282 pdus (with sense 4), R2T 547 pdus
      Data-in 140543388 pdus, 189570075420 bytes
```

The **show iscsi stats** command can be used to view brief or detailed iSCSI statistics per iSCSI interface. See Examples [22-22](#) and [22-23](#).

### *Example 22-22 Displays iSCSI Statistics for the iSCSI Interface*

```
switch# show iscsi stats iscsi 2/1
iscsi2/1
  5 minutes input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
```

```

5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
iSCSI statistics
  3568 packets input, 1134600 bytes
    Command 2930 pdus, Data-out 471 pdus, 939008 bytes, 0 fragments
  output 13418 packets, 10785796 bytes
    Response 2930 pdus (with sense 18), R2T 235 pdus
  Data-in 10086 pdus, 10134572 bytes

```

### Example 22-23 Displays Detailed iSCSI Statistics for the iSCSI Interface

```

switch# show iscsi stats iscsi 2/1 detail
iscsi2/1
  5 minutes input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec

```



**Note** The above line displays the throughput on the iSCSI side in the inbound direction.

```

5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec

```



**Note** The above line displays the throughput on the iSCSI side in the outbound direction.

```

iSCSI statistics
  3568 packets input, 1134600 bytes

```



**Note** The above line displays the number of packets and bytes received from the iSCSI side.

```

Command 2930 pdus, Data-out 471 pdus, 939008 bytes, 0 fragments

```



**Note** The above line displays the number of commands, write data PDUs, write data bytes, and fragmented data PDUs received from the iSCSI side)

```

output 13418 packets, 10785796 bytes

```



**Note** The above line displays the number of packets and bytes sent to the iSCSI side.

```

Response 2930 pdus (with sense 18), R2T 235 pdus

```



**Note** The above line displays the number of response PDUs (with sense information) and number of R2T PDUs sent to the iSCSI side.

```

Data-in 10086 pdus, 10134572 bytes

```



**Note** The above line displays the number of read data PDUs and bytes sent to the iSCSI side.

```

iSCSI Forward:
  Command: 2930 PDUs (Rcvd: 2930)

```



**Note** The above line displays the number of command PDUs forwarded to the FC side (out of the number of command PDU received from iSCSI side.

```
Data-Out (Write): 471 PDUs (Rcvd 471), 0 fragments, 939008 bytes
```

**Note**

The above line displays the number of write data PDUs forwarded to the FC side (out of the number of write data PDUs received from iSCSI side), fragmented data PDUs forwarded to the FC side from the iSCSI side, data bytes forwarded to the FC side from the iSCSI side.

```
FCP Forward:
  Xfer_rdy: 235 (Rcvd: 240)
```

**Note**

The above line displays the number of transfer-ready frames forwarded to the iSCSI side (out of the number of transfer-ready frames received from the FC side).

```
Data-In: 10086 (Rcvd: 10093), 10134572 bytes
```

**Note**

The above line displays the number of read data frames forwarded to the iSCSI side (out of the number of read data frames received from the FC side))

```
Response: 2930 (Rcvd: 2942), with sense 18
```

**Note**

The above line displays the number of response frames forwarded to the iSCSI side (out of the number of response frames received from the FC side), number of response frames with sense bytes forwarded to iSCSI side from the FC side)

```
TMF Resp: 0
```

**Note**

The above line displays the number of Task Management Function (TMF) response frames forwarded to the iSCSI side from the FC side.

```
iSCSI Stats:
  Login: attempt: 40, succeed: 9, fail: 31, authen fail: 0
```

**Note**

The above line displays the number of logins attempted, succeeded, failed, and those that failed due to authentication problems.

```
Rcvd: NOP-Out: 13, Sent: NOP-In: 13
```

**Note**

The above line displays the number of No Operation Opcode (NOP)-Out PDUs received, number of NOP-In PDUs sent.

```
NOP-In: 0, Sent: NOP-Out: 0
```

**Note**

The above line displays the number of NOP-In PDUs received, number of NOP-Out PDUs sent.

```
TMF-REQ: 0, Sent: TMF-RESP: 0
```

**Note**

The above line displays the number of TMF requests received, number of TMF responses sent.

---

Text-REQ: 2, Sent: Text-RESP: 2

**Note**

The above line displays the number of Text requests (Text-REQ) received, number of Text responses (Text-RESP) sent.

---

SNACK: 0

**Note**

The above line displays the number of Selective Negative Acknowledgment (SNACK) PDUs received.

---

Unrecognized Opcode: 0, Bad header digest: 0

**Note**

The above line displays the number of unknown opcode PDUs received and the number of PDUs received with bad header digest.

---

Command in window but not next: 0, exceed wait queue limit: 0

**Note**

The above line displays the number of commands received that did not have the next expected sequence number, but were in the allowable window—these commands were placed on the wait queue. The second counter displays the number of times the wait queue limit was exceeded.

---

Received PDU in wrong phase: 0

**Note**

The above line displays the number of PDUs received in the wrong phase of iSCSI connection.

---

SCSI Busy responses: 0

**Note**

The above line displays the number of SCSI Busy responses sent to the iSCSI side.

---

FCP Stats:

Total: Sent: 3733 (number of FCP frames sent out on the FC side)

Received: 13373 (Error: 0, Unknown: 0)

**Note**

The above line displays the number of FCP frames received from the FC side.

---

Sent: PLOGI: 38, Rcvd: PLOGI\_ACC: 38, PLOGI\_RJT: 0

**Note**

The above line displays the number of PLOGIs sent, PLOGI accepts, and PLOGI rejects received.

---

PRLI: 100, Rcvd: PRLI\_ACC: 7, PRLI\_RJT: 0, Error: 0, From initiator: 0

**Note**

The above line displays the number of Process LogIn (PRLI) sent, PRLI accepts, and PRLI rejects received, number of PRLI responses received with error, number of PRLI responses received from an initiator)

---

LOGO: 35, Rcvd: LOGO\_ACC: 0, LOGO\_RJT: 0

**Note**

The above line displays the number of log outs (LOGOs) sent, LOGO accepts, and LOGO rejects received.

---

```
PRLO: 4, Rcvd: PRLO_ACC: 0, PRLO_RJT: 0
```

**Note**

The above line displays the number of Process LogOuts (PRLOs) sent, PRLO accepts, and PRLO rejects received.

```
ABTS: 93, Rcvd: ABTS_ACC: 0
```

**Note**

The above line displays the number of ABort TaskS (ABTS) sent and the ABTS accepts received.

```
TMF REQ: 0
```

**Note**

The above line displays the number of TMF requests sent.

```
Self orig command: 7, Rcvd: data: 7, resp: 7
```

**Note**

The above line displays the number of IPS module originated commands sent, data and responses received.

```
Rcvd: PLOGI: 35, Sent: PLOGI_ACC: 33, PLOGI_RJT: 2
```

**Note**

The above line displays the number of PLOGI received, PLOGI\_ACC and PLOGI\_RJT sent.

```
LOGO: 4, Sent: LOGO_ACC: 4, LOGO_RJT: 0
```

**Note**

The above line displays the number of LOGO received, LOGO\_ACC and LOGO\_RJT sent.

```
PRLI: 6, Sent: PRLI_ACC: 6, PRLI_RJT: 0
```

**Note**

The above line displays the number of PRLI received, PRLI\_ACC and PRLI\_RJT sent.

```
PRLO: 0, Sent: PRLO_ACC: 0, PRLO_RJT: 0
```

**Note**

The above line displays the number of PRLO received, PRLO\_ACC and PRLO\_RJT sent.

```
ABTS: 0
```

**Note**

The above line displays the number of ABTS received.

```
iSCSI Drop:
  Command: Target down 0, Task in progress 0, LUN map fail 0
           CmdSeqNo not in window 0, No Exchange ID 0, Reject 0
           No task: 0
```

**Note**

The above line displays the commands dropped due to the FC target being down, tasks already in progress, iSCSI to FC LUN mapping failures, command sequence number being outside the allowable window, no FC exchange ID s available in the internal table, supported commands which were rejected, and new tasks that could not be allocated.



```
Data-Out: 0, Data CRC Error: 0
```

**Note**

The above line displays the number of data out PDUs dropped and the number of data out PDUs dropped due to data CRC errors.

```
TMF-Req: 0, No task: 0
```

**Note**

The above line displays the TMF requests dropped due to various reasons (cumulative of: new task could not be allocated, FC exchange id table full, iSCSI to FC lun mapping failed, command sequence number outside the allowable window), TMF responses dropped because no task was available and new one could not be allocated.

```
FCP Drop:
Xfer_rdy: 0, Data-In: 0, Response: 0
```

**Note**

The above line displays the number of dropped transfer-ready, read data packets, and responses from the FC side.

```
Buffer Stats:
Buffer less than header size: 0, Partial: 477, Split: 237
Pullup give new buf: 0, Out of contiguous buf: 0, Unaligned m_data: 0
```

**Note**

The above lines display the buffer related statistics for internal use.

## Displaying Proxy Initiator Information

If the proxy initiator feature is enabled in the iSCSI interface, use the **show interface iscsi** command to display configured proxy initiator information (see Examples 22-24 and 22-25).

### *Example 22-24 Displays Proxy Initiator Information for the iSCSI Interface with System-Assigned WWNs*

```
switch# show interface iscsi 4/1
iscsi4/1 is up
  Hardware is GigabitEthernet
  Port WWN is 20:c1:00:05:30:00:a7:9e
  Admin port mode is iSCSI
  Port mode is iSCSI
  Speed is 1 Gbps
  iSCSI initiator is identified by name
  Number of iSCSI session: 0, Number of TCP connection: 0
  Configured TCP parameters
    Local Port is 3260
    PMTU discover is enabled, reset timeout is 3600 sec
    Keepalive-timeout is 60 sec
    Minimum-retransmit-time is 300 ms
    Max-retransmissions 4
    Sack is disabled
    QOS code point is 0
  Forwarding mode: pass-thru
  TMF Queueing Mode : disabled
  Proxy Initiator Mode : enabled<-----Proxy initiator is enabled
    nWWN is 28:00:00:05:30:00:a7:a1 (system-assigned)<----System-assigned nWWN
    pWWN is 28:01:00:05:30:00:a7:a1 (system-assigned)<---- System-assigned pWWN
  5 minutes input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
```

```

5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
iSCSI statistics
  Input 7 packets, 2912 bytes
    Command 0 pdus, Data-out 0 pdus, 0 bytes
  Output 7 packets, 336 bytes
    Response 0 pdus (with sense 0), R2T 0 pdus
    Data-in 0 pdus, 0 bytes

```

**Example 22-25 Displays Proxy Initiator Information for the iSCSI Interface with User-Assigned WWNs**

```

switch# show interface iscsi 4/2
iscsi4/2 is up
  Hardware is GigabitEthernet
  Port WWN is 20:c1:00:05:30:00:a7:9e
  Admin port mode is ISCSI
  Port mode is ISCSI
  Speed is 1 Gbps
  iSCSI initiator is identified by name
  Number of iSCSI session: 0, Number of TCP connection: 0
  Configured TCP parameters
    Local Port is 3260
    PMTU discover is enabled, reset timeout is 3600 sec
    Keepalive-timeout is 60 sec
    Minimum-retransmit-time is 300 ms
    Max-retransmissions 4
    Sack is disabled
    QOS code point is 0
  Forwarding mode: pass-thru
  TMF Queueing Mode : disabled
  Proxy Initiator Mode : enabled
    nWWN is 11:11:11:11:11:11:11:11 (manually-configured)<----User-assigned nWWN
    pWWN is 22:22:22:22:22:22:22:22 (manually-configured)<----User-assigned pWWN
  5 minutes input rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
  5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
  iSCSI statistics
    Input 7 packets, 2912 bytes
      Command 0 pdus, Data-out 0 pdus, 0 bytes
    Output 7 packets, 336 bytes
      Response 0 pdus (with sense 0), R2T 0 pdus
      Data-in 0 pdus, 0 bytes

```

## Displaying Global iSCSI Information

Use the **show iscsi global** command to view the overall configuration and the iSCSI status. See [Example 22-26](#).

**Example 22-26 Displays the Current Global iSCSI Configuration and State**

```

switch# show iscsi global
iSCSI Global information
  Authentication: NONE
  Import FC Target: Enabled
  Number of target nodes: 5
  Number of portals: 8
  Number of sessions: 6
  Failed session: 0, Last failed initiator name:

```

## Displaying iSCSI Sessions

Use the **show iscsi session** command to view details about the current iSCSI sessions in the switch. Without parameters, this command displays all sessions. The output can be filtered by specifying an initiator, a target, or both.

[Example 22-27](#) displays one iSCSI initiator configured based on the IQN name (iqn.1987-05.com.cisco:02.3021b0f2fda0.avanti12-w2k) and another based on its IP address (10.10.100.199).

### *Example 22-27 Displays Brief Information of All iSCSI Sessions*

```
switch# show iscsi session
Initiator iqn.1987-05.com.cisco:02.3021b0f2fda0.avanti12-w2k
  Initiator ip addr (s): 10.10.100.116
  Session #1
    Discovery session, ISID 00023d000043, Status active

  Session #2
    Target VT1
    VSAN 1, ISID 00023d000046, Status active, no reservation

  Session #3
    Target VT2
    VSAN 1, ISID 00023d000048, Status active, no reservation

Initiator 10.10.100.199
  Initiator name iqn.1987-05.com.cisco.01.7e3183ae458a94b1cd6bc168cba09d2e
  Session #1
    Target VT2
    VSAN 1, ISID 246700000000, Status active, no reservation

  Session #2
    Target VT1
    VSAN 1, ISID 246b00000000, Status active, no reservation

  Session #3
    Target iqn.1987-05.com.cisco:05.switch.04-01.2100002037a6be32
    VSAN 1, ISID 246e00000000, Status active, no reservation
```

[Example 22-28](#) and [Example 22-29](#) display the iSCSI initiator configured based on its IP address (10.10.100.199).

### *Example 22-28 Displays Brief Information About the Specified iSCSI Session*

```
switch# show iscsi session initiator 10.10.100.199 target VT1
Initiator 10.10.100.199
  Initiator name iqn.1987-05.com.cisco.01.7e3183ae458a94b1cd6bc168cba09d2e
  Session #1
    Target VT1
    VSAN 1, ISID 246b00000000, Status active, no reservation
```

### *Example 22-29 Displays Detailed Information About the Specified iSCSI Session*

```
switch# show iscsi session initiator 10.10.100.199 target VT1 detail
Initiator 10.10.100.199 (oasis-qa)
  Initiator name iqn.1987-05.com.cisco.01.7e3183ae458a94b1cd6bc168cba09d2e
  Session #1 (index 3)
    Target VT1
```

```

VSAN 1, ISID 246b00000000, TSIH 384, Status active, no reservation
Type Normal, ExpCmdSN 39, MaxCmdSN 54, Barrier 0
MaxBurstSize 0, MaxConn 0, DataPDUInOrder No
DataSeqInOrder No, InitialR2T Yes, ImmediateData No
Registered LUN 0, Mapped LUN 0
Stats:
  PDU: Command: 38, Response: 38
  Bytes: TX: 8712, RX: 0
Number of connection: 1
Connection #1
  Local IP address: 10.10.100.200, Peer IP address: 10.10.100.199
  CID 0, State: LOGGED_IN
  StatSN 62, ExpStatSN 0
  MaxRecvDSLength 1024, our_MaxRecvDSLength 1392
  CSG 3, NSG 3, min_pdu_size 48 (w/ data 48)
  AuthMethod none, HeaderDigest None (len 0), DataDigest None (len 0)
  Version Min: 2, Max: 2
  FC target: Up, Reorder PDU: No, Marker send: No (int 0)
  Received MaxRecvDSLLen key: No

```

## Displaying iSCSI Initiators

Use the **show iscsi initiator** command to display information about all initiators connected to an iSCSI interface in the switch. The information can be filtered to display only the desired iSCSI initiator by specifying the initiator name. Detailed output of the iscsi initiator can be obtained by specifying the **detail** option. The **iscsi-session** (and optionally **detail**) parameter displays only iSCSI session information. The **fcp-session** (and optionally **detail**) parameter displays only FCP session information. The output includes static and dynamic initiators. See Examples 22-30 and 22-31.

### Example 22-30 Displays Information About Connected iSCSI Initiators

```

switch# show iscsi initiator
iSCSI Node name is iqn.1987-05.com.cisco:02.3021b0f2fda0.avanti12-w2k
  Initiator ip addr (s): 10.10.100.116
  iSCSI alias name: AVANTI12-W2K
  Node WWN is 22:01:00:05:30:00:10:e1 (configured)
  Member of vsans: 1, 2, 10
  Number of Virtual n_ports: 1
  Virtual Port WWN is 22:04:00:05:30:00:10:e1 (configured)
    Interface iSCSI 4/1, Portal group tag: 0x180
    VSAN ID 1, FCID 0x6c0202
    VSAN ID 2, FCID 0x6e0000
    VSAN ID 10, FCID 0x790000

iSCSI Node name is 10.10.100.199
  iSCSI Initiator name: iqn.1987-05.com.cisco.01.7e3183ae458a94b1cd6bc168cba09d2e
  iSCSI alias name: oasis-qa
  Node WWN is 22:03:00:05:30:00:10:e1 (configured)
  Member of vsans: 1, 5
  Number of Virtual n_ports: 1
  Virtual Port WWN is 22:00:00:05:30:00:10:e1 (configured)
    Interface iSCSI 4/1, Portal group tag: 0x180
    VSAN ID 5, FCID 0x640000
    VSAN ID 1, FCID 0x6c0203

```

### Example 22-31 Displays Detailed Information About the iSCSI Initiator

```

switch# show iscsi initiator iqn.1987-05.com.cisco:02.3021b0f2fda0.avanti12-w2k detail
iSCSI Node name is iqn.1987-05.com.cisco:02.3021b0f2fda0.avanti12-w2k
  Initiator ip addr (s): 10.10.100.116

```

```

iSCSI alias name: AVANTI12-W2K
Node WWN is 22:01:00:05:30:00:10:e1 (configured)
Member of vsans: 1, 2, 10
Number of Virtual n_ports: 1

Virtual Port WWN is 22:04:00:05:30:00:10:e1 (configured)
Interface iSCSI 4/1, Portal group tag is 0x180
VSAN ID 1, FCID 0x6c0202
1 FC sessions, 1 iSCSI sessions
iSCSI session details          <-----iSCSI session details
  Target: VT1
  Statistics:
    PDU: Command: 0, Response: 0
    Bytes: TX: 0, RX: 0
    Number of connection: 1
  TCP parameters
    Local 10.10.100.200:3260, Remote 10.10.100.116:4190
    Path MTU: 1500 bytes
    Retransmission timeout: 310 ms
    Round trip time: Smoothed 160 ms, Variance: 38
    Advertized window: Current: 61 KB, Maximum: 62 KB, Scale: 0
    Peer receive window: Current: 63 KB, Maximum: 63 KB, Scale: 0
    Congestion window: Current: 1 KB

FCP Session details          <-----FCP session details
  Target FCID: 0x6c01e8 (S_ID of this session: 0x6c0202)
  pWWN: 21:00:00:20:37:62:c0:0c, nWWN: 20:00:00:20:37:62:c0:0c
  Session state: CLEANUP
  1 iSCSI sessions share this FC session
  Target: VT1
  Negotiated parameters
    RcvDataFieldSize 1392 our_RcvDataFieldSize 1392
    MaxBurstSize 0, EMPD: FALSE
    Random Relative Offset: FALSE, Sequence-in-order: Yes
  Statistics:
    PDU: Command: 0, Response: 0

```

Use the **show fcns database** (and optionally **detail**) to display the Fibre Channel name server entry for the Fibre Channel N port created for iSCSI initiators in the SAN. See Examples 22-32 and 22-35.

### Example 22-32 Displays the FCNS Database Contents

```

switch# show fcns database
VSAN 1:
-----
FCID          TYPE  PWWN                               (VENDOR)          FC4-TYPE:FEATURE
-----
0x020101      N     22:04:00:05:30:00:35:e1 (Cisco)           scsi-fcp:init isc..w <--iSCSI
0x020102      N     22:02:00:05:30:00:35:e1 (Cisco)           scsi-fcp:init isc..w initiator
0x0205d4      NL    21:00:00:04:cf:da:fe:c6 (Seagate)         scsi-fcp:target
0x0205d5      NL    21:00:00:04:cf:e6:e4:4b (Seagate)         scsi-fcp:target
0x0205d6      NL    21:00:00:04:cf:e6:21:ac (Seagate)         scsi-fcp:target
0x0205d9      NL    21:00:00:04:cf:e6:19:9b (Seagate)         scsi-fcp:target
0x0205da      NL    21:00:00:04:cf:e6:19:62 (Seagate)         scsi-fcp:target
0x0205dc      NL    21:00:00:04:cf:e6:e9:82 (Seagate)         scsi-fcp:target
0x0205e0      NL    21:00:00:04:cf:e6:21:06 (Seagate)         scsi-fcp:target
0x0205e1      NL    21:00:00:04:cf:e6:e0:eb (Seagate)         scsi-fcp:target

Total number of entries = 10

VSAN 2:
-----
FCID          TYPE  PWWN                               (VENDOR)          FC4-TYPE:FEATURE
-----

```

```
-----
0xef0001    N    22:02:00:05:30:00:35:e1 (Cisco)    scsi-fcp:init isc..w
```

```
Total number of entries = 1
```

```
VSAN 3:
```

```
-----
FCID        TYPE  PWWN                (VENDOR)    FC4-TYPE:FEATURE
-----
0xed0001    N    22:02:00:05:30:00:35:e1 (Cisco)    scsi-fcp:init isc..w
```

```
Total number of entries = 1
```

### Example 22-33 Displays the FCNS Database in Detail

```
switch# show fcns database detail
-----
VSAN:1      FCID:0x020101
-----
port-wwn (vendor)    :22:04:00:05:30:00:35:e1 (Cisco)
node-wwn             :22:03:00:05:30:00:35:e1
class                :2,3
node-ip-addr         :10.2.2.12
ipa                  :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name   :
symbolic-node-name   :iqn.1991-05.com.microsoft:oasis2-dell
port-type            :N
port-ip-addr         :0.0.0.0
fabric-port-wwn     :22:01:00:05:30:00:35:de
hard-addr            :0x000000
-----
VSAN:1      FCID:0x020102
-----
port-wwn (vendor)    :22:02:00:05:30:00:35:e1 (Cisco)
node-wwn             :22:01:00:05:30:00:35:e1
class                :2,3
node-ip-addr         :10.2.2.11
ipa                  :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name   :
symbolic-node-name   :iqn.1987-05.com.cisco.01.14ac33ba567f986f174723b5f9f2377
port-type            :N
port-ip-addr         :0.0.0.0
fabric-port-wwn     :22:01:00:05:30:00:35:de
hard-addr            :0x000000
...
Total number of entries = 10
=====
-----
VSAN:2      FCID:0xef0001
-----
port-wwn (vendor)    :22:02:00:05:30:00:35:e1 (Cisco)
node-wwn             :22:01:00:05:30:00:35:e1
class                :2,3
node-ip-addr         :10.2.2.11
ipa                  :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name   :
symbolic-node-name   :iqn.1987-05.com.cisco.01.14ac33ba567f986f174723b5f9f2377
port-type            :N
port-ip-addr         :0.0.0.0
```

```
fabric-port-wwn      :22:01:00:05:30:00:35:de
hard-addr            :0x000000
```

```
Total number of entries = 1
```

```
...
```

**Example 22-34 Displays Detailed Information for a Fibre Channel N Port Created for An iSCSI Initiator Identified by Its IQN Name**

```
switch# show fcns database fcid 0x6c0203 detail vsan 1
-----
VSAN:1      FCID:0x6c0203
-----
port-wwn (vendor)   :22:00:00:05:30:00:10:e1 (Cisco)
node-wwn            :22:03:00:05:30:00:10:e1
class               :2,3
node-ip-addr        :10.10.100.199
ipa                 :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name  :
symbolic-node-name  :10.10.100.199
port-type           :N
port-ip-addr        :0.0.0.0
fabric-port-wwn     :20:c1:00:05:30:00:10:de
hard-addr           :0x000000

Total number of entries = 1
```

**Example 22-35 Displays Detailed Information for a Fibre Channel N Port Created for An iSCSI Initiator Identified by Its IP Address**

```
switch# show fcns database fcid 0x6c0203 detail vsan 1
-----
VSAN:1      FCID:0x6c0203
-----
port-wwn (vendor)   :22:00:00:05:30:00:10:e1 (Cisco)
node-wwn            :22:03:00:05:30:00:10:e1
class               :2,3
node-ip-addr        :10.10.100.199
ipa                 :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name  :
symbolic-node-name  :10.10.100.199<---ID assigned by IP address
port-type           :N
port-ip-addr        :0.0.0.0
fabric-port-wwn     :20:c1:00:05:30:00:10:de
hard-addr           :0x000000

Total number of entries = 1
```

Use the **show iscsi initiator configured** to display information about all the configured iSCSI initiators. Specifying the name shows information about the desired initiator. See [Example 22-36](#).

**Example 22-36 Display Information About Configured Initiators**

```
switch# show iscsi initiator configured
iSCSI Node name is iqn.1987-05.com.cisco:02.3021b0f2fda0.avanti12-w2k
Member of vsans: 1, 2, 10
Node WWN is 22:01:00:05:30:00:10:e1
No. of PWWN: 5
```

```

Port WWN is 22:04:00:05:30:00:10:e1
Port WWN is 22:05:00:05:30:00:10:e1
Port WWN is 22:06:00:05:30:00:10:e1
Port WWN is 22:07:00:05:30:00:10:e1
Port WWN is 22:08:00:05:30:00:10:e1

iSCSI Node name is 10.10.100.199
Member of vsans: 1, 5
Node WWN is 22:03:00:05:30:00:10:e1
No. of PWWN: 4
Port WWN is 22:00:00:05:30:00:10:e1
Port WWN is 22:09:00:05:30:00:10:e1
Port WWN is 22:0a:00:05:30:00:10:e1
Port WWN is 22:0b:00:05:30:00:10:e1

```

## Displaying iSCSI Virtual Targets

Use the **show iscsi virtual-target** to display information about the FC targets exported as iSCSI virtual targets to the iSCSI initiators. The output includes static as well as dynamic targets. See [Example 22-37](#).

### Example 22-37 Displays Exported Targets

```

switch# show iscsi virtual-target
target: VT1
* Port WWN 21:00:00:20:37:62:c0:0c
  Configured node
  all initiator permit is enabled

target: VT2
  Port WWN 21:00:00:04:cf:4c:52:c1
  Configured node
  all initiator permit is disabled

target: iqn.1987-05.com.cisco:05.switch.04-01.2100002037a6be32
  Port WWN 21:00:00:20:37:a6:be:32 , VSAN 1
  Auto-created node

```

## Displaying IPS Statistics

The **show ips stats tcp interface** command displays information about the underlying transport for iSCSI. See Examples [22-38](#) and [22-39](#).

### Example 22-38 Displays iSCSI Stats (In Brief)

```

switch# show ips stats tcp interface gigabitethernet 2/1
TCP Statistics for port GigabitEthernet2/1
  Connection Stats
    0 active openings, 6 accepts
    0 failed attempts, 0 reset received, 6 established
  Segment stats
    640780835 received, 150953931 sent, 12 retransmitted
    0 bad segments received, 0 reset sent
  TCP Active Connections

```

Local Address	Remote Address	State	Send-Q	Recv-Q
10.48.69.250:3260	10.48.69.226:1026	ESTABLISH	0	0
10.48.69.250:3260	10.48.69.231:1026	ESTABLISH	0	0
10.48.69.250:3260	10.48.69.231:1033	ESTABLISH	0	0
10.48.69.250:3260	10.48.69.226:1038	ESTABLISH	0	0
0.0.0.0:3260	0.0.0.0:0	LISTEN	0	0



**Example 22-39 Displays SCSI Stats (In Detail)**

```

switch# show ips stats tcp interface gigabitethernet 2/1 detail
TCP Statistics for port GigabitEthernet2/1
  TCP send stats
    150953931 segments, 2755572300 bytes
    53986369 data, 82341597 ack only packets
    4 control (SYN/FIN/RST), 0 probes, 14625949 window updates
    12 segments retransmitted, 576 bytes
    12 retransmitted while on ethernet send queue, 0 packets split
    118741734 delayed acks sent
  TCP receive stats
    640780835 segments, 640325552 data packets in sequence, 925034009772 bytes in
sequence
    0 predicted ack, 615117910 predicted data
    0 bad checksum, 0 multi/broadcast, 0 bad offset
    0 no memory drops, 0 short segments
    0 duplicate bytes, 0 duplicate packets
    0 partial duplicate bytes, 0 partial duplicate packets
    0 out-of-order bytes, 0 out-of-order packets
    0 packet after window, 0 bytes after window
    0 packets after close
    25656078 acks, 2755572210 ack bytes, 0 ack toomuch, 5786 duplicate acks
    0 ack packets left of snd_una, 0 non-4 byte aligned packets
    12100 window updates, 0 window probe
    29 pcb hash miss, 17 no port, 0 bad SYN, 0 paws drops
  TCP Connection Stats
    0 attempts, 6 accepts, 6 established
    4 closed, 4 drops, 0 conn drops
    0 drop in retransmit timeout, 4 drop in keepalive timeout
    0 drop in persist drops, 0 connections drained
  TCP Miscellaneous Stats
    21635776 segments timed, 21642712 rtt updated
    12 retransmit timeout, 0 persist timeout
    8494 keepalive timeout, 8490 keepalive probes
  TCP SACK Stats
    0 recovery episodes, 0 data packets, 0 data bytes
    0 data packets retransmitted, 0 data bytes retransmitted
    0 connections closed, 0 retransmit timeouts
  TCP SYN Cache Stats
    6 entries, 6 connections completed, 0 entries timed out
    0 dropped due to overflow, 0 dropped due to RST
    0 dropped due to ICMP unreachable, 0 dropped due to bucket overflow
    0 abort due to no memory, 0 duplicate SYN, 0 no-route SYN drop
    0 hash collisions, 0 retransmitted
  TCP Active Connections
    Local Address      Remote Address      State      Send-Q  Recv-Q
    10.48.69.250:3260  10.48.69.226:1026  ESTABLISH  0       0
    10.48.69.250:3260  10.48.69.231:1026  ESTABLISH  0       0
    10.48.69.250:3260  10.48.69.231:1033  ESTABLISH  0       0
    10.48.69.250:3260  10.48.69.226:1038  ESTABLISH  0       0
    0.0.0.0:3260       0.0.0.0:0          LISTEN     0       0

```

The **show ips stats buffer** command displays information about the iSCSI buffers. See Example 22-40.

**Example 22-40 Displays iSCSI Buffers**

```

switch# show ips stats buffer interface gigabitethernet 4/2
Mbuf Statistics for port GigabitEthernet4/2
Free Mbufs           : 83221
Mbuf high watermark  : 124830
Mbuf low watermark   : 20805
Mbuf alloc failures  : 0

```

```

Total clusters           : 2304
Free Clusters           : 80145
Clusters high watermark : 87381
Clusters low watermark  : 79059
Clusters alloc failures : 0
Free shared mbufs       : 0
Shared Mbuf alloc failures : 0
Free shared clusters    : 0
Shared clusters alloc failures: 0

Ether channel Statistics for port GigabitEthernet4/2
TCP segments sent       : 0
TCP segments received   : 0
Xmit packets sent       : 0
Xmit packets received   : 0
Config packets sent     : 0
Config packets received : 0
MPQ packets send errors : 0

```

## Displaying iSCSI User Information

The **show user-account iscsi** command displays all configured iSCSI user names. See [Example 22-41](#).

### *Example 22-41 Displays iSCSI User Names*

```

switch# show user-account iscsi
username:iscsiuser
secret: dsfffsffsffasffsdfg

username:user2
secret:cshadhdsadadjajdjas

```

## iSCSI High Availability

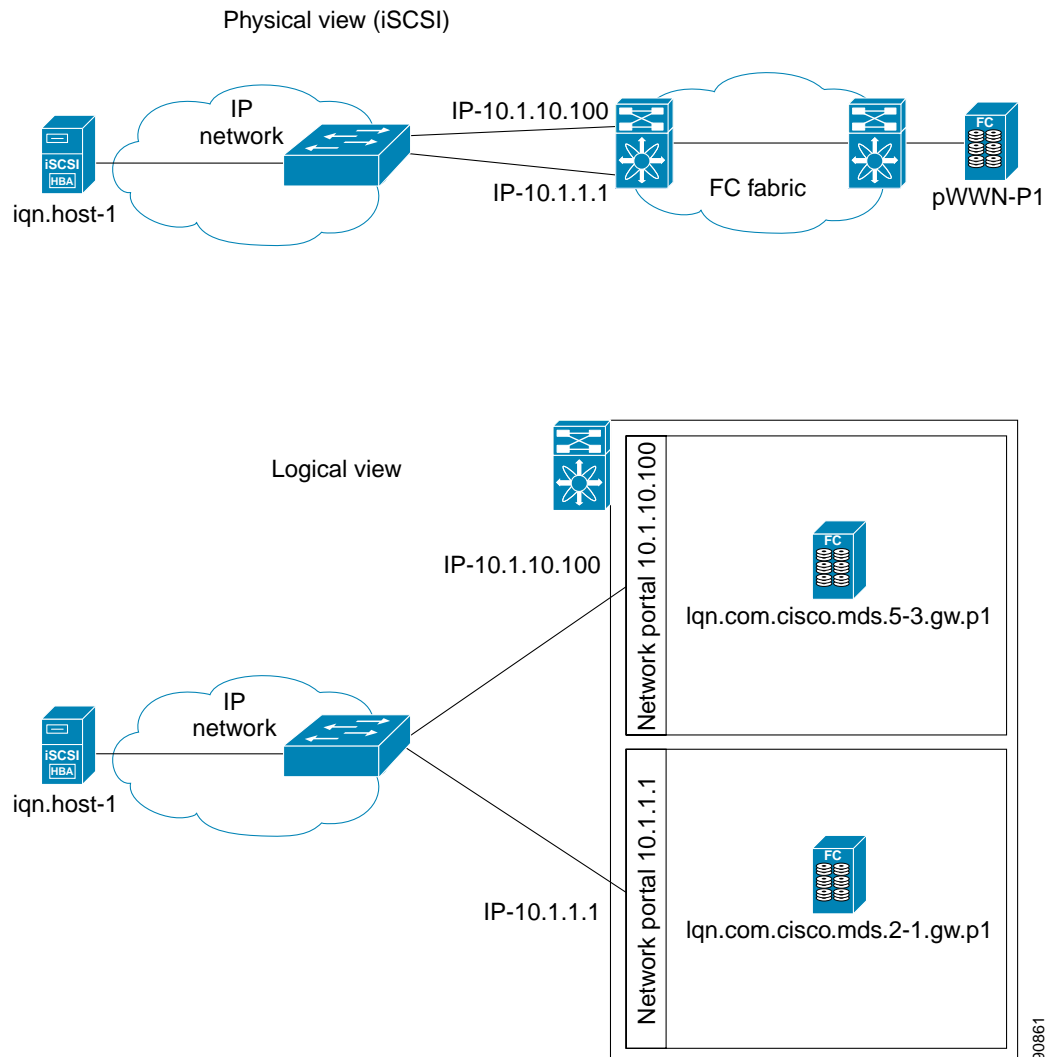
The following high availability features are available for iSCSI configurations:

- [Multiple IPS Ports Connected to the Same IP Network, page 22-83](#)
- [VRRP-Based High Availability, page 22-84](#)
- [Ethernet PortChannel-Based High Availability, page 22-85](#)

## Multiple IPS Ports Connected to the Same IP Network

Figure 22-31 provides an example of a configuration with multiple Gigabit Ethernet interfaces in the same IP network.

Figure 22-31 Multiple Gigabit Ethernet Interfaces in the Same IP Network

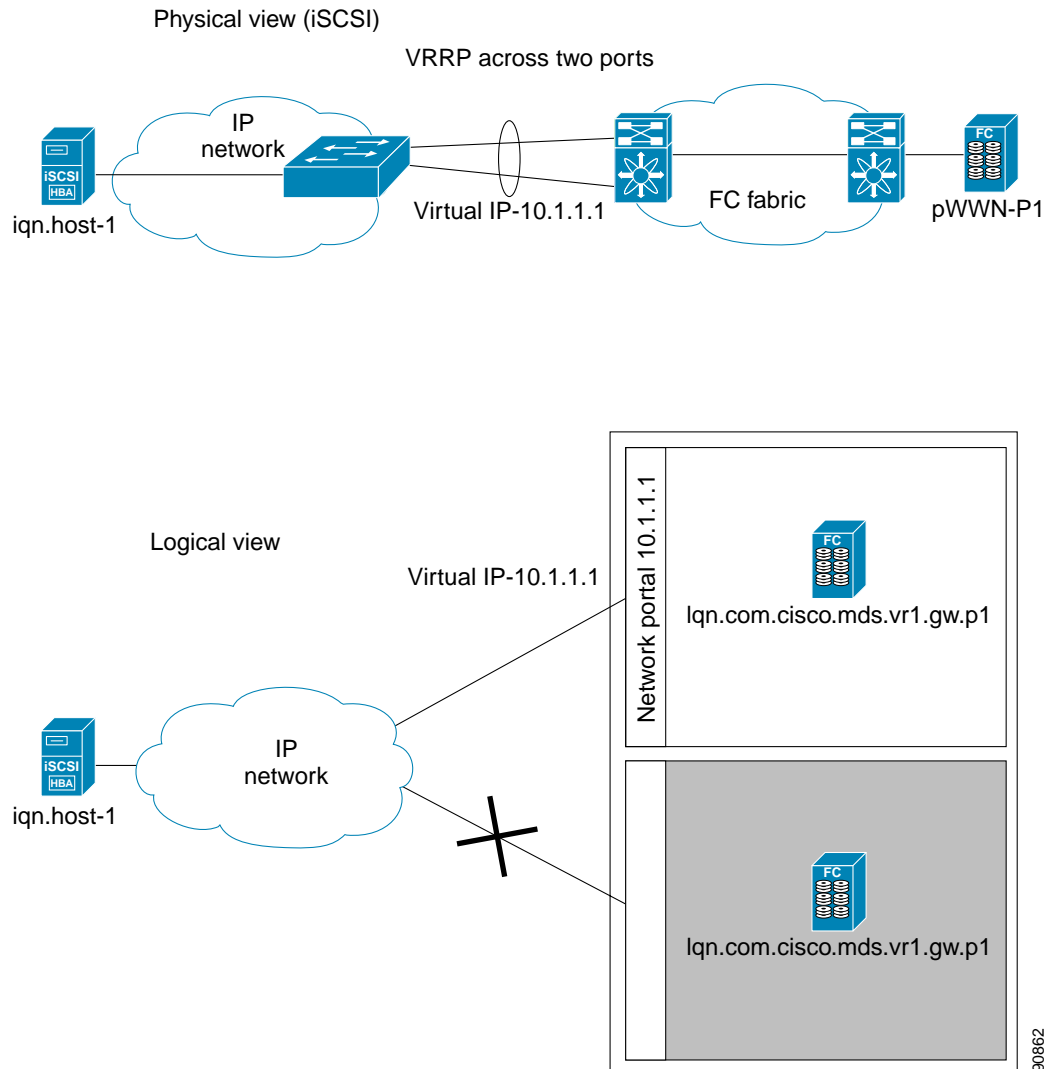


In Figure 22-31, each iSCSI host discovers two iSCSI targets for every physical Fibre Channel target (with different names). The multi-pathing software on the host provides load-balancing over both paths. If one Gigabit Ethernet interface fails, the host multi-pathing software is not affected because it can use the second path.

## VRRP-Based High Availability

Figure 22-32 provides an example of a VRRP-based high availability iSCSI configuration.

Figure 22-32 VRRP-Based iSCSI High Availability



In Figure 22-32, each iSCSI host discovers one iSCSI target for every physical Fibre Channel target. When the Gigabit Ethernet interface of the VRRP master fails, the iSCSI session is terminated. The host then reconnects to the target and the session comes up because the second Gigabit Ethernet interface has taken over the virtual IP address as the new master.



### Tip

Ports that act as VRRP master and backup can be on different switches. If you have a static WWN configuration for iSCSI initiators (see the [“Presenting iSCSI Hosts as Virtual Fibre Channel Hosts”](#) section on page 22-55), configure a different WWN for the iSCSI initiator for each switch. If you use a proxy initiator, be sure to configure a different pWWN on each iSCSI interface for each VRRP port used.

## Ethernet PortChannel-Based High Availability

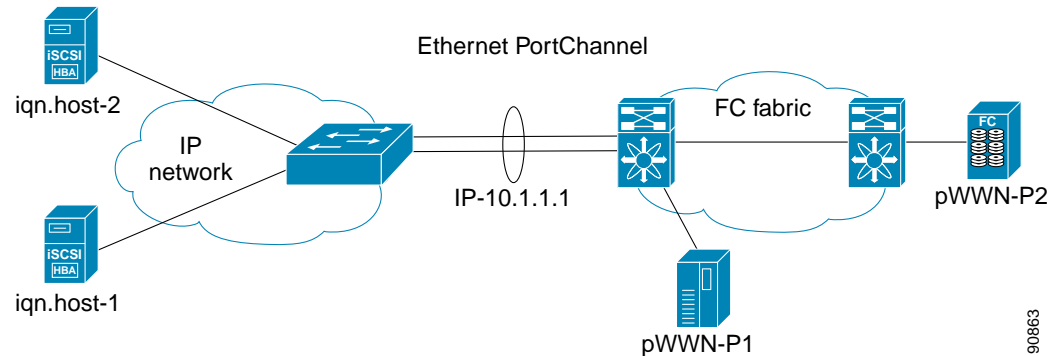


Note

All iSCSI data traffic for one iSCSI link is carried on one TCP connection. Consequently, the aggregated bandwidth is 1 Gbps for that iSCSI link.

Figure 22-33 provides a sample Ethernet PortChannel-based high availability iSCSI configuration.

Figure 22-33 Ethernet PortChannel-Based iSCSI High Availability



In Figure 22-33, each iSCSI host discovers one iSCSI target for every physical Fibre Channel target. The iSCSI session from the iSCSI host to the virtual iSCSI target (on the IPS port) uses one of the two physical interfaces (because an iSCSI session uses one TCP connection). When the Gigabit Ethernet interface fails, the IPS module and the Ethernet switch transparently forwards all the frames on to the second Gigabit Ethernet interface.

## iSCSI Setup Guidelines and Scenarios

This section provides guidelines on iSCSI authentication possibilities, setup requirements, and sample scenarios. It includes the following authentication setup guidelines:

- [No Authentication, page 22-85](#)
- [CHAP with Local Password Database, page 22-86](#)
- [CHAP with External RADIUS Server, page 22-86](#)



Note

This section does not specify the steps to enter or exit EXEC mode, configuration mode, or any submode. Be sure to verify the prompt before issuing any command.

### No Authentication

Set the iSCSI authentication method to **none** to configure a network with no authentication.

```
switch(config)# iscsi authentication none
```

## CHAP with Local Password Database

To configure authentication using the CHAP option with the local password database, follow these steps:

- Step 1** Set the AAA authentication to use the local password database for iSCSI protocol.

```
switch(config)# aaa authentication iscsi default local
```

- Step 2** Set the iSCSI authentication method to require CHAP for all iSCSI clients.

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

- Step 3** Configure the user names and passwords for iSCSI users.

```
switch(config)# username iscsi-user password abcd iscsi
```



**Note** If you do not specify the **iscsi** option, the user name is assumed to be a Cisco MDS switch user instead of an iSCSI user.

- Step 4** Verify the global iSCSI authentication setup.

```
switch# show iscsi global
iSCSI Global information Authentication: CHAP <----Verify
  Import FC Target: Disabled
  ...
```

## CHAP with External RADIUS Server

To configure authentication using the CHAP option with an external RADIUS server, follow these steps:

- Step 1** Configure the password for the Cisco MDS switch as RADIUS client to the RADIUS server.

```
switch(config)# radius-server key mds-1
```

- Step 2** Configure the RADIUS server IP address.

```
switch(config)# radius-server host 10.1.1.10
```

- Step 3** Configure a server group.

```
switch(config)# aaa group server radius iscsi-radius-group
switch(config-radius)# server 10.1.1.1
```

- Step 4** Set up the authentication verification for the iSCSI protocol to go to the RADIUS server.

```
switch(config)# aaa authentication iscsi default group iscsi-radius-group
```

- Step 5** Set up the iSCSI authentication method to require CHAP for all iSCSI clients.

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

- Step 6** Verify that the global iSCSI authentication set up is CHAP.

```
switch# show iscsi global
iSCSI Global information
  Authentication: CHAP <----- Verify CHAP
  ....
```

**Step 7** Verify that the AAA authentication information for iSCSI.

```
switch# show aaa authentication
      default: local
      console: local
      iscsi: group iscsi-radius-group   <----- Group name
      dhchap: local

switch# show radius-server groups
total number of groups:2

following RADIUS server groups are configured:
  group radius:
    server: all configured radius servers
  group iscsi-radius-group:
    server: 10.1.1.1 on auth-port 1812, acct-port 1813

switch# show radius-server
Global RADIUS shared secret:mds-1   <----- Verify secret
....

following RADIUS servers are configured:
  10.1.1.1:   <----- Verify the server IP address
    available for authentication on port:1812
    available for accounting on port:1813
```

---

To configure an iSCSI RADIUS server, follow these steps:

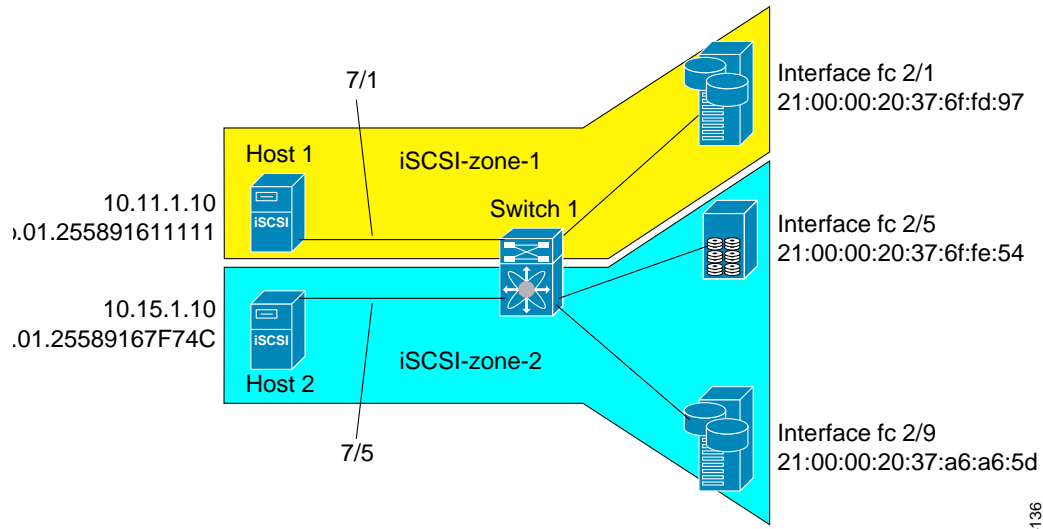
- 
- Step 1** Configure the RADIUS server to allow access from the Cisco MDS switch's management Ethernet IP address.
  - Step 2** Configure the shared secret for the RADIUS server to authenticate the Cisco MDS switch.
  - Step 3** Configure the iSCSI users and passwords on the RADIUS server.
- 

## Scenario 1

Sample scenario 1 assumes the following configuration (see [Figure 22-34](#)):


- There is no access control using Fibre Channel zoning.
- There is no target-based LUN mapping or LUN masking.
- There is no iSCSI authentication (none).
- The iSCSI initiator is identified using IP address (host 1 = 10.11.1.10).
- The iSCSI initiator is identified using node name (host 2 = iqn.1987-05.com.cisco:01.25589167f74c).

Figure 22-34 iSCSI Scenario 1



94136

To configure scenario 1 (see Figure 22-34), follow these steps:

- 
- Step 1** Configure null authentication for all iSCSI hosts.
- ```
switch(config)# iscsi authentication none
```
- Step 2** Configure iSCSI to dynamically import all FC targets into the iSCSI SAN using auto-generated iSCSI target names.
- ```
switch(config)# iscsi import target fc
```
- Step 3** Configure the Gigabit Ethernet interface in slot 7 port 1 with an IP address and enable the interface.
- ```
switch(config)# int gigabitethernet 7/1
switch(config-if)# ip address 10.11.1.1 255.255.255.0
switch(config-if)# no shut
```
-  **Note** Host 1 is connected to this port.
- 
- Step 4** Configure the iSCSI interface in slot 7 port 1 to identify all dynamic iSCSI initiators by the IP address, and enable the interface.
- ```
switch(config)# int iscsi 7/1
switch(config-if)# switchport initiator id ip-address
switch(config-if)# no shut
```
- Step 5** Configure the Gigabit Ethernet interface in slot 7 port 5 with the IP address and enable the interface.
- ```
switch(config)# int gigabitethernet 7/5
switch(config-if)# ip address 10.15.1.1 255.255.255.0
switch(config-if)# no shut
```
- Step 6** Configure the iSCSI interface in slot 7 port 5 to identify all dynamic iSCSI initiators by node name and enable the interface.
- ```
switch(config)# int iscsi 7/5
switch(config-if)# switchport initiator id name
```



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



**Note** Host 1 is connected to this port.

**Step 7** Verify the available Fibre Channel targets (see [Figure 22-34](#)).

```
switch# show fcns database
VSAN 1:
```

```
-----
FCID          TYPE  PWWN                                (VENDOR)          FC4-TYPE:FEATURE
-----
0x6d0001      NL    21:00:00:20:37:6f:fd:97 (Seagate)         scsi-fcp:target
0x6d0101      NL    21:00:00:20:37:6f:fe:54 (Seagate)         scsi-fcp:target
0x6d0201      NL    21:00:00:20:37:a6:a6:5d (Seagate)         scsi-fcp:target
Total number of entries = 3
```

**Step 8** Create a zone named *iscsi-zone-1* with host 1 and one Fibre Channel target in it.



**Note** Use the IP address of the host in zone membership configuration because the iSCSI interface is configured to identify all hosts based on IP address.

```
switch(config)# zone name iscsi-zone-1 vsan 1
switch(config-zone)# member pwn 21:00:00:20:37:6f:fd:97
switch(config-zone)# member symbolic-nodename 10.11.1.10
```

**Step 9** Create a zone named *iscsi-zone-2* with host 2 and two FC targets in it.



**Note** Use the symbolic node name of the iSCSI host in zone membership configuration because the iSCSI interface is configured to identify all hosts based on node name.

```
switch(config)# zone name iscsi-zone-2 vsan 1
switch(config-zone)# member pwn 21:00:00:20:37:6f:fe:54
switch(config-zone)# member pwn 21:00:00:20:37:a6:a6:5d
switch(config-zone)# member symbolic-nodename iqn.1987-05.com.cisco:01.25589167f74c
```

**Step 10** Create a zone set and add the two zones as members.

```
switch(config)# zoneset name zoneset-iscsi vsan 1
switch(config-zoneset)# member iscsi-zone-1
switch(config-zoneset)# member iscsi-zone-2
```

**Step 11** Activate the zone set.

```
switch(config)# zoneset activate name zoneset-iscsi vsan 1
```

**Step 12** Display the active zone set.



**Note** The iSCSI hosts has not connected so they do not have a FCID yet.

```
switch# show zoneset active
zoneset name zoneset-iscsi vsan 1
  zone name iscsi-zone-1 vsan 1
    * fcid 0x6d0001 [pwn 21:00:00:20:37:6f:fd:97] <-----Target
      symbolic-nodename 10.11.1.10 <-----iSCSI host (host 1, not online)

  zone name iscsi-zone-2 vsan 1
    * fcid 0x6d0101 [pwn 21:00:00:20:37:6f:fe:54] <-----Target
```

```
* fcid 0x6d0201 [pwwn 21:00:00:20:37:a6:a6:5d] <-----Target
symbolic-nodename iqn.1987-05.com.cisco:01.25589167f74c <-iSCSI host (host 2, not online)
```

**Step 13** Bring up the iSCSI hosts (host 1 and host 2).

**Step 14** Show all the iSCSI sessions (use the **detail** option for detailed information).

```
switch# show iscsi session
Initiator iqn.1987-05.com.cisco:01.25589167f74c <-----Host 2
Initiator ip addr (s): 10.15.1.11
Session #1
Target iqn.1987-05.com.cisco:05.172.22.92.166.07-05.21000020376ffe54
```



**Note** The last part of the auto-created target name is the FC target's pWWN.

```
VSAN 1, ISID 00023d000001, Status active, no reservation
```

```
Session #2
Target iqn.1987-05.com.cisco:05.172.22.92.166.07-05.2100002037a6a65d
VSAN 1, ISID 00023d000001, Status active, no reservation
```

```
Initiator 10.11.1.10 <-----Host 1
Initiator name iqn.1987-05.com.cisco:01.e41695d16b1a
Session #1
Target iqn.1987-05.com.cisco:05.172.22.92.166.07-01.21000020376ffd97
VSAN 1, ISID 00023d000001, Status active, no reservation
```

**Step 15** Verify the details of the two iSCSI initiators.

```
switch# show iscsi initiator
iSCSI Node name is iqn.1987-05.com.cisco:01.25589167f74c <-----
Initiator ip addr (s): 10.15.1.11
iSCSI alias name: oasis11.cisco.com
Node WWN is 20:02:00:0b:fd:44:68:c2 (dynamic)
Member of vsans: 1
Number of Virtual n_ports: 1
Virtual Port WWN is 20:03:00:0b:fd:44:68:c2 (dynamic)
Interface iSCSI 7/5, Portal group tag: 0x304
VSAN ID 1, FCID 0x6d0300
```

**Host 2: Initiator ID based on node name because the initiator is entering iSCSI interface 7/5**

```
iSCSI Node name is 10.11.1.10 <-----
iSCSI Initiator name: iqn.1987 - 05.com.cisco:01.e41695d16b1a
iSCSI alias name: oasis10.cisco.com
Node WWN is 20:04:00:0b:fd:44:68:c2 (dynamic)
Member of vsans: 1
Number of Virtual n_ports: 1
Virtual Port WWN is 20:05:00:0b:fd:44:68:c2 (dynamic)
Interface iSCSI 7/1, Portal group tag: 0x300
VSAN ID 1, FCID 0x6d0301
```

**Host 1: Initiator ID based on IP address because the initiator is entering iSCSI interface 7/1**

**Step 16** View the active zone set. The iSCSI initiators' FCIDs are resolved.

```
switch# show zoneset active
zoneset name zoneset-iscsi vsan 1
  zone name iscsi-zone-1 vsan 1
    * fcid 0x6d0001 [pwwn 21:00:00:20:37:6f:fd:97]
    * fcid 0x6d0301 [symbolic-nodename 10.11.1.10] <-----
```

**FCID resolved for host 1**

```

  zone name iscsi-zone-2 vsan 1
    * fcid 0x6d0101 [pwwn 21:00:00:20:37:6f:fe:54]
    * fcid 0x6d0201 [pwwn 21:00:00:20:37:a6:a6:5d]
    * fcid 0x6d0300 [symbolic-nodename
iqn.1987-05.com.cisco:01.25589167f74c]<-----
```

**FCID for host 2**

**Step 17** The Fibre Channel name server shows the virtual N ports created for the iSCSI hosts.

```
switch# show fcns database
VSAN 1:
-----
FCID          TYPE  PWWN                               (VENDOR)      FC4-TYPE:FEATURE
-----
0x6d0001     NL   21:00:00:20:37:6f:fd:97 (Seagate)     scsi-fcp:target
0x6d0101     NL   21:00:00:20:37:6f:fe:54 (Seagate)     scsi-fcp:target
0x6d0201     NL   21:00:00:20:37:a6:a6:5d (Seagate)     scsi-fcp:target
0x6d0300     N    20:03:00:0b:fd:44:68:c2 (Cisco)        scsi-fcp:init isc..w
0x6d0301     N    20:05:00:0b:fd:44:68:c2 (Cisco)        scsi-fcp:init isc..w
```

**Step 18** Verify the detailed output of the iSCSI initiator nodes in the Fibre Channel name server.

```
switch# show fcns database fcid 0x6d0300 detail vsan 1
-----
VSAN:1      FCID:0x6d0300
-----
port-wnn (vendor)      :20:03:00:0b:fd:44:68:c2 (Cisco)
node-wnn              :20:02:00:0b:fd:44:68:c2
class                 :2,3
node-ip-addr          :10.15.1.11 <-----
ipa                   :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw <-----
symbolic-port-name    :
```

**IP address of the iSCSI host**

**iSCSI gateway node**

```

symbolic-node-name
:iqn.1987-05.com.cisco:01.25589167f74c<-----
port-type              :N
port-ip-addr          :0.0.0.0
fabric-port-wnn       :21:91:00:0b:fd:44:68:c0
hard-addr              :0x000000
Total number of entries = 1
```

**iSCSI initiator ID is based on the registered node name**

```

switch# show fcns database fcid 0x6d0301 detail vsan 1
-----
VSAN:1      FCID:0x6d0301
-----
port-wwn (vendor)      :20:05:00:0b:fd:44:68:c2 (Cisco)
node-wwn               :20:04:00:0b:fd:44:68:c2
class                 :2,3
node-ip-addr           :10.11.1.10
ipa                   :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw <-----
symbolic-port-name     :

symbolic-node-name     :10.11.1.10 <-----
port-type              :N
port-ip-addr           :0.0.0.0
fabric-port-wwn       :21:81:00:0b:fd:44:68:c0
hard-addr              :0x000000

```

iSCSI gateway node

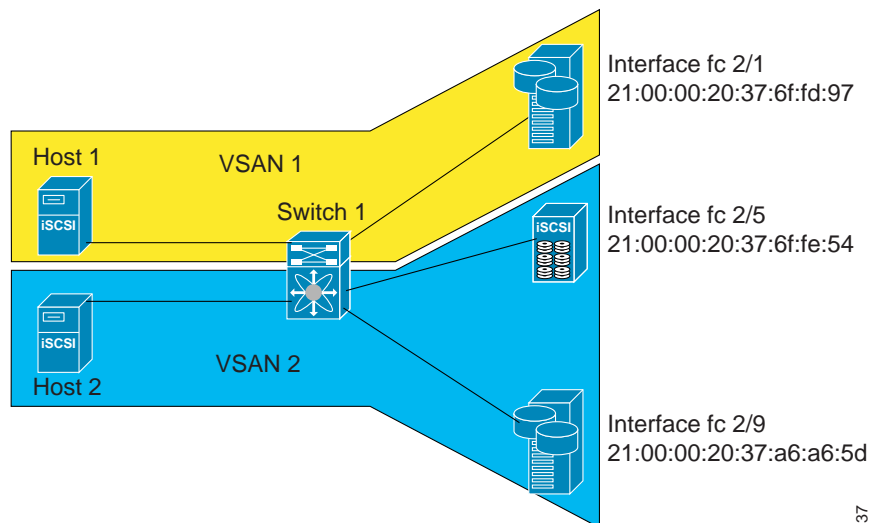
iSCSI initiator ID is based on the IP address registered in symbolic-node-name field

## Scenario 2

Sample scenario 2 assumes the following configuration (see [Figure 22-35](#)):

- Access control is based on Fibre Channel zoning.
- There is target-based LUN mapping or LUN masking.
- There is no iSCSI authentication (none).
- The iSCSI initiator is assigned to different VSANs.

**Figure 22-35 iSCSI Scenario 2**



94137

To configure scenario 2 (see [Figure 22-35](#)), follow these steps:

**Step 1** Configure null authentication for all iSCSI hosts.

```
switch(config)# iscsi authentication none
```

- Step 2** Configure iSCSI to dynamically import all FC targets into the iSCSI SAN using auto-generated iSCSI target names.

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

- Step 3** Configure the Gigabit Ethernet interface in slot 7 port 1 with an IP address and enable the interface.

```
switch(config)# int gigabitethernet 7/1
switch(config-if)# ip address 10.11.1.1 255.255.255.0
switch(config-if)# no shut
```

- Step 4** Configure the iSCSI interface in slot 7 port 1 to identify all dynamic iSCSI initiators by the IP address and enable the interface.

```
switch(config)# int iscsi 7/1
switch(config-if)# switchport initiator id ip-address
switch(config-if)# no shut
```

- Step 5** Configure the Gigabit Ethernet interface in slot 7 port 5 with the IP address and enable the interface.

```
switch(config)# int gigabitethernet 7/5
switch(config-if)# ip address 10.15.1.1 255.255.255.0
switch(config-if)# no shut
```

- Step 6** Configure the iSCSI interface in slot 7 port 5 to identify all dynamic iSCSI initiators by IP address and enable the interface.

```
switch(config)# int iscsi 7/5
switch(config-if)# switchport initiator id name
switch(config-if)# no shut
```

- Step 7** Add static configuration for each iSCSI initiator.

```
switch(config)# iscsi initiator name iqn.1987-05.com.cisco:01.e41695d16b1a <----Host 2
switch(config-iscsi-init)# static pwwn system-assign 1
switch(config-iscsi-init)# static nwwn system-assign

switch(config)# iscsi initiator ip address 10.15.1.11
switch(config-iscsi-init)# static pwwn system-assigned 1
switch(config-iscsi-init)# vsan 2
```

- Step 8** View the configured initiators.




---

**Note** The WWNs are assigned by the system. The initiators are members of different VSANs.

---

```
switch# show iscsi initiator configured
iSCSI Node name is iqn.1987-05.com.cisco:01.e41695d16b1a
  Member of vsans: 1
  Node WWN is 20:03:00:0b:fd:44:68:c2
  No. of PWWN: 1
  Port WWN is 20:02:00:0b:fd:44:68:c2

iSCSI Node name is 10.15.1.11
  Member of vsans: 2
  No. of PWWN: 1
  Port WWN is 20:06:00:0b:fd:44:68:c2
```

- Step 9** Create a zone with host 1.

```
switch(config)# zone name iscsi-zone-1 vsan 1
```

- Step 10** Add three members to the zone named *iscsi-zone-1*.



**Note** Fibre Channel storage for zone membership for the iSCSI initiator, either the iSCSI symbolic node name or the pWWN, can be used. In this case, the pWWN is persistent.

- The following command is based on the symbolic node name.

```
switch(config-zone)# member symbolic-nodename iqn.1987-05.com.cisco:01.e41695d16b1a
```

- The following command is based on the persistent pWWN assigned to the initiator. You can obtain the pWWN from the **show iscsi initiator** output.

```
switch(config-zone)# member pwn 20:02:00:0b:fd:44:68:c2
```

**Step 11** Create a zone with host 2 and two Fibre Channel targets.



**Note** If the host is in VSAN 2, the Fibre Channel targets and zone must also be in VSAN 2.

**Step 12** Activate the zone set in VSAN 2

```
switch(config)# zone name iscsi-zone-2 vsan 2
switch(config)# zoneset activate name iscsi-zoneset-v2 vsan 2
Zoneset activation initiated. check zone status
switch# show zoneset active vsan 2
zoneset name iscsi-zoneset-v2 vsan 2
zone name iscsi-zone-2 vsan 2
* fcid 0x750001 [pwn 21:00:00:20:37:6f:fe:54]
* fcid 0x750101 [pwn 21:00:00:20:37:a6:a6:5d]
pwn 20:06:00:0b:fd:44:68:c2 <-----Host is not online
```

**Step 13** Start the iSCSI clients on both hosts and verify that sessions come up.

**Step 14** Display the iSCSI sessions to verify the Fibre Channel target and the configured WWNs.

```
switch# show iscsi session
Initiator iqn.1987-05.com.cisco:01.e41695d16b1a
Initiator ip addr (s): 10.11.1.10
Session #1
Discovery session, ISID 00023d000001, Status active

Session #2
Target
iqn.1987-05.com.cisco:05.172.22.92.166.07-01.21000020376ffd97<---- To FC target
VSAN 1, ISID 00023d000001, Status active, no reservation
```

**Step 15** Display the iSCSI initiator to verify the configured nWWN and pWWN.

```
switch# show iscsi initiator
iSCSI Node name is iqn.1987-05.com.cisco:01.e41695d16b1a
Initiator ip addr (s): 10.11.1.10
iSCSI alias name: oasis10.cisco.com

Node WWN is 20:03:00:0b:fd:44:68:c2 (configured)<----- The configured nWWN
Member of vsans: 1
Number of Virtual n_ports: 1

Virtual Port WWN is 20:02:00:0b:fd:44:68:c2 (configured)<---- The configured pWWN
Interface iSCSI 7/1, Portal group tag: 0x300
VSAN ID 1, FCID 0x680102
```

**Step 16** Check the Fibre Channel name server.

```
switch# show fcns database vsan 1
VSAN 1:
-----
FCID      TYPE PWWN                               (VENDOR) FC4-TYPE:FEATURE
-----
0x680001 NL 21:00:00:20:37:6f:fd:97 (Seagate) scsi-fcp:target

0x680102 N 20:02:00:0b:fd:44:68:c2 (Cisco) scsi-fcp:init isc..w <---
```

**iSCSI initiator in  
name server****Step 17** Verify the details of the iSCSI initiator's FCID in the name server.

```
switch(config)# show fcns database fcid 0x680102 detail vsan 1
-----
VSAN:1      FCID:0x680102
-----
port-wwn (vendor)      :20:02:00:0b:fd:44:68:c2 (Cisco)
node-wwn              :20:03:00:0b:fd:44:68:c2
class                 :2,3
node-ip-addr          :10.11.1.10
ipa                   :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name    :
symbolic-node-name    :iqn.1987-05.com.cisco:01.e41695d16b1a
port-type             :N
port-ip-addr          :0.0.0.0
fabric-port-wwn       :21:81:00:0b:fd:44:68:c0
    iSCSI alias name: oasis10.cisco.com
```

```
Node WWN is 20:03:00:0b:fd:44:68:c2 (configured)<-----
Member of vsans: 1
Number of Virtual n_ports: 1
```

**The configured nWWN**

```
Virtual Port WWN is 20:02:00:0b:fd:44:68:c2 (configured)<---
Interface iSCSI 7/1, Portal group tag: 0x300
VSAN ID 1, FCID 0x680102
```

**The configured pWWN****Step 18** Check the Fibre Channel name server

```
switch# show fcns database vsan 1
VSAN 1:
-----
FCID      TYPE PWWN                               (VENDOR) FC4-TYPE:FEATURE
-----
0x680001 NL 21:00:00:20:37:6f:fd:97 (Seagate) scsi-fcp:target

0x680102 N 20:02:00:0b:fd:44:68:c2 (Cisco) scsi-fcp:init isc..w <-----
```

**iSCSI  
initiator in  
name server****Step 19** Verify the details of the iSCSI initiator's FCID in the name server.

```
switch(config)# show fcns database fcid 0x680102 detail vsan 1
-----
VSAN:1      FCID:0x680102
```

```

-----
port-wwn (vendor)      :20:02:00:0b:fd:44:68:c2 (Cisco)
node-wwn              :20:03:00:0b:fd:44:68:c2
class                 :2,3
node-ip-addr          :10.11.1.10
ipa                   :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name    :
symbolic-node-name    :iqn.1987-05.com.cisco:01.e41695d16b1a
port-type             :N
port-ip-addr          :0.0.0.0
fabric-port-wwn      :21:81:00:0b:fd:44:68:c0
hard-addr             :0x000000

```

**Step 20** Verify that zoning has resolved the FCID for the iSCSI client.

```

switch# show zoneset active vsan 1
zoneset name iscsi-zoneset-v1 vsan 1
  zone name iscsi-zone-1 vsan 1
    * fcid 0x680001 [pwwn 21:00:00:20:37:6f:fd:97]
    * fcid 0x680102 [pwwn 20:02:00:0b:fd:44:68:c2]

```

**Step 21** Verify that the second initiator is connected to the two Fibre Channel targets in VSAN 2.

```

switch# show iscsi session initiator 10.15.1.11
Initiator 10.15.1.11
  Initiator name iqn.1987-05.com.cisco:01.25589167f74c
  Session #1
    Target iqn.1987-05.com.cisco:05.172.22.92.166.07-05.21000020376ffe54 <-- Session to
    VSAN 2, ISID 00023d000001, Status active, no reservation                first target

  Session #2
    Target iqn.1987-05.com.cisco:05.172.22.92.166.07-05.2100002037a6a65d <-- Session to
    VSAN 2, ISID 00023d000001, Status active, no reservation                second
                                                                              target

switch# show iscsi initiator
iSCSI Node name is 10.15.1.11 <--- Initiator ID is the IP address
  iSCSI Initiator name: iqn.1987-05.com.cisco:01.25589167f74c
  iSCSI alias name: oasis11.cisco.com

  Node WWN is 20:04:00:0b:fd:44:68:c2 (dynamic) <----- Dynamic
  Member of vsans: 2 <--- vsan membership                    WWN as
  Number of Virtual n_ports: 1                                static WWN
                                                                              not
                                                                              assigned

  Virtual Port WWN is 20:06:00:0b:fd:44:68:c2 (configured) <----- Static
  Interface iSCSI 7/5, Portal group tag: 0x304                pWWN for
  VSAN ID 2, FCID 0x750200                                    the initiator

```

```

switch# show fcns database vsan 2
VSAN 2:
-----
FCID      TYPE  PWWN                                (VENDOR)  FC4-TYPE:FEATURE
-----
0x750001  NL    21:00:00:20:37:6f:fe:54 (Seagate) scsi-fcp:target
0x750101  NL    21:00:00:20:37:a6:a6:5d (Seagate) scsi-fcp:target

```



```
0x750200 N 20:06:00:0b:fd:44:68:c2 (Cisco) scsi-fcp:init isc..w <--
Total number of entries = 3
```

**iSCSI  
initiator  
entry in  
name server**

```
switch# show fcns database fcid 0x750200 detail vsan 2
```

```
-----
VSAN:2 FCID:0x750200
-----
port-wwn (vendor) :20:06:00:0b:fd:44:68:c2 (Cisco)
node-wwn :20:04:00:0b:fd:44:68:c2
class :2,3
node-ip-addr :10.15.1.11
ipa :ff ff ff ff ff ff ff ff
fc4-types:fc4_features:scsi-fcp:init iscsi-gw
symbolic-port-name :
symbolic-node-name :10.15.1.11
port-type :N
port-ip-addr :0.0.0.0
fabric-port-wwn :21:91:00:0b:fd:44:68:c0
hard-addr :0x000000
Total number of entries = 1
```

```
switch# show zoneset active vsan 2
```

```
zoneset name iscsi-zoneset-v2 vsan 2
  zone name iscsi-zone-2 vsan 2
    * fcid 0x750001 [pwwn 21:00:00:20:37:6f:fe:54]
    * fcid 0x750101 [pwwn 21:00:00:20:37:a6:a6:5d]

    * fcid 0x750200 [pwwn 20:06:00:0b:fd:44:68:c2] <-----
```

**FCID  
resolved for  
iSCSI  
initiator**

# Configuring Storage Name Services

As of Cisco MDS SAN-OS Release 1.3(1), the Internet Storage Name Service (iSNS) client feature is available in all switches in the Cisco MDS 9000 Family with IPS modules installed.

iSNS services allow your existing TCP/IP networks to function more effectively as storage area networks by automating the discovery, management, and configuration of iSCSI devices. To facilitate these functions, the iSNS client functionality registers iSCSI portals and all targets accessible through a particular interface with an external iSNS server.

## Creating iSNS Profiles and Tagging Profiles

The iSNS client functionality on each interface (Gigabit Ethernet interface or subinterface or PortChannel) registers information with its configured iSNS server using an iSNS profile. This process is referred to as tagging an iSNS profile to an interface. Each iSNS profile keeps information about an iSNS server IP address. One profile can be tagged to one or more interfaces.

Once a profile is tagged to an interface, the MDS switch opens a TCP connection to the iSNS server IP address (using a well-known iSNS port number 3205) in the profile and registers network entity and portal objects. It goes through the FC name server database and configuration to find storage nodes to register with the server.

Statically mapped virtual targets are registered if the associated Fibre channel pWWN is present in the FC name server database and no access control configuration prevents it (using the **advertise interface** or the **initiator** options in the **iscsi virtual-target name** command). A dynamically mapped target is registered if the dynamic target importing is enabled using the **iscsi import target fc** command.

See the “[Presenting Fibre Channel Targets as iSCSI Targets](#)” section on page 22-48.

A storage node is deregistered from the iSNS server when it becomes unavailable either because of configuration changes (such as access control change or dynamic import disabling) or when the Fibre Channel storage port goes off-line. It is registered again when the node is online.

When the iSNS client is unable to register/deregister objects with the iSNS server (for example, the client is unable to make a TCP connection to the iSNS server), it retries every minute to re-register all iSNS objects for the affected interface(s) with the iSNS server.

Untagging a profile causes the network entity and portal to deregister from that interface.

## Creating an iSNS Profile

To create an iSNS profile, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>isns profile name MyIsns</b> switch(config-(isns-profile))#	Creates a profile called MyIsns.
	switch(config)# <b>no isns profile name OldIsns</b>	Removes a configured iSNS profile called OldIsns.
Step 3	switch(config-(isns-profile))# <b>server 10.10.100.211</b>	Specifies an iSNS server IP address for this profile.
	switch(config-(isns-profile))# <b>no server 10.20.100.211</b>	Removes a configured iSNS server from this profile.

## Modifying an iSNS Profile

To modify (tag) the iSNS profile for an interface, untag the interface from currently tagged profile and then tag to a new profile

To modify the iSNS profile for a profile, remove the existing server and then add the new server.

To tag an interface to a profile, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface gigabitethernet 4/1</b> switch(config-if)#	Configures the specified Gigabit Ethernet interface.
Step 3	switch(config-if)# <b>isns MyIsns</b>	Tags this interface to the profile.

To untag an interface from a profile, follow these steps:

	Command	Purpose
Step 1	switch# <b>config t</b> switch(config)#	Enters configuration mode.
Step 2	switch(config)# <b>interface gigabitethernet 5/1</b> switch(config-if)#	Configures the specified Gigabit Ethernet interface.
Step 3	switch(config-if)# <b>no isns OldIsns</b>	Untags this interface from the profile.

All associated iSNS objects for an interface tagged to an iSNS profile can be re-registered with the iSNS server using the **isns reregister** command in EXEC mode.

```
switch# isns reregister gigabitethernet 1/4
switch# isns reregister port-channel 1
switch#
```

## Verifying iSNS Configurations

Use the **show isns profile** command to view configured iSNS profiles. Profile ABC has two portals registered with the iSNS server. Each portal corresponds to a particular interface. Profile XYZ has a specified iSNS server, but does not have any tagged interfaces configured (see Examples 22-42 and 22-43).

### Example 22-42 Displays all Configured iSNS Profiles

```
switch# show isns profile
iSNS profile name ABC
tagged interface GigabitEthernet2/3
tagged interface GigabitEthernet2/2
iSNS Server 10.10.100.204

iSNS profile name XYZ
iSNS Server 10.10.100.211
```

### Example 22-43 Displays a Specified iSNS Profile

```
switch# show isns profile ABC
```

```
iSNS profile name ABC
tagged interface GigabitEthernet2/3
tagged interface GigabitEthernet2/2
iSNS Server 10.10.100.204
```

Use the **show isns profile counters** command to view all configured profiles with the iSNS PDU statistics for each tagged interface (see Examples 22-44 and 22-45).

#### **Example 22-44 Displays Configured Profiles with iSNS Statistics**

```
switch# show isns profile counters
iSNS profile name ABC
tagged interface port-channel 1
iSNS statistics
  Input 54 pdus (registration/deregistration pdus only)
    Reg pdus 37, Dereg pdus 17
  Output 54 pdus (registration/deregistration pdus only)
    Reg pdus 37, Dereg pdus 17
iSNS Server 10.10.100.204

iSNS profile name XYZ
tagged interface port-channel 2
iSNS statistics
  Input 30 pdus (registration/deregistration pdus only)
    Reg pdus 29, Dereg pdus 1
  Output 30 pdus (registration/deregistration pdus only)
    Reg pdus 29, Dereg pdus 1
iSNS Server 10.1.4.218
```

#### **Example 22-45 Displays a Specified Profile's iSNS Statistics**

```
switch# show isns profile ABC counters
iSNS profile name ABC
tagged interface port-channel 1
iSNS statistics
  Input 54 pdus (registration/deregistration pdus only)
    Reg pdus 37, Dereg pdus 17
  Output 54 pdus (registration/deregistration pdus only)
    Reg pdus 37, Dereg pdus 17
iSNS Server 10.10.100.204
```

Use the **show isns** command to view all objects registered on the iSNS server and specified in the given profile (see Example 22-46).

#### **Example 22-46 Displays iSNS Queries**

```
switch# show isns query ABC gigabitethernet 2/3
iSNS server: 10.10.100.204
Init: iqn.1991-05.com.microsoft:ibmw2k
  Alias: <MS SW iSCSI Initiator>
Tgt : iqn.1987-05.com.cisco:05.172.22.94.22.02-03
Tgt : iqn.1987-05.com.cisco:05.172.22.94.22.02-03.210000203762fa34
  nWWN: 200000203762fa34
```

Use the **show interface** command to view the iSNS profile to which an interface is tagged (see [Example 22-47](#)).

**Example 22-47 Displays Tagged iSNS Interfaces**

```
switch# show int gigabitethernet 2/3
GigabitEthernet2/3 is up
Hardware is GigabitEthernet, address is 0005.3000.ae94
Internet address is 10.10.100.201/24
MTU 1500 bytes
Port mode is IPS
Speed is 1 Gbps
Beacon is turned off
Auto-Negotiation is turned on
iSNS profile ABC
^^^^^^^^^^^^^^^^^^

5 minutes input rate 112 bits/sec, 14 bytes/sec, 0 frames/sec
5 minutes output rate 0 bits/sec, 0 bytes/sec, 0 frames/sec
1935 packets input, 132567 bytes
 4 multicast frames, 0 compressed
 0 input errors, 0 frame, 0 overrun 0 fifo
1 packets output, 42 bytes, 0 underruns
 0 output errors, 0 collisions, 0 fifo
 0 carrier errors
```

## Default Settings

[Table 22-2](#) lists the default settings for Gigabit Ethernet parameters.

**Table 22-2 Default Gigabit Ethernet Parameters**

Parameters	Default
IP MTU frame size	1500 bytes for all Ethernet ports
Auto-negotiation	Enabled.
Promiscuous mode	Disabled

[Table 22-3](#) lists the default settings for FCIP parameters.

**Table 22-3 Default FCIP Parameters**

Parameters	Default
TCP default port for FCIP	3225.
minimum-retransmit-time	200 ms.
keepalive-timeout	60 seconds.
max-retransmissions	4 retransmissions.
PMTU discovery	Enabled.
pmtu-enable reset-timeout	3600 seconds.
SACK	Enabled.
max-bandwidth	1Gbps.

**Table 22-3 Default FCIP Parameters (continued)**

Parameters	Default
min-available-bandwidth	500 Mbps.
round-trip-time	1 ms.
buffer size	0 KB.
Control TCP and data connection	No packets are transmitted.
TCP congestion window monitoring	Enabled.
Burst size	10KB.
TCP connection mode	Active mode is enabled.
special-frame	Disabled.
FCIP timestamp	Disabled.
acceptable-diff range to accept packets	+ or - 1000 ms.
B port keepalive responses	Disabled.

Table 22-4 lists the default settings for iSCSI parameters.

**Table 22-4 Default iSCSI Parameters**

Parameters	Default
Number of TCP connections	One per iSCSI session.
minimum-retransmit-time	200 ms.
keepalive-timeout	60 seconds.
max-retransmissions	4 retransmissions.
PMTU discovery	Enabled.
pmtu-enable reset-timeout	3600 seconds.
SACK	Enabled.
max-bandwidth	1G.bps
min-available-bandwidth	70 Mbps.
round-trip-time	1 ms.
buffer size	4096 KB.
Control TCP and data connection	No packets are transmitted.
TCP congestion window monitoring	Enabled.
Burst size	50KB.
TCP connection mode	Active mode is enabled.
Fibre Channel targets to iSCSI	Not imported.
Advertising iSCSI target	Advertised on all Gigabit Ethernet interfaces, subinterfaces, PortChannel interfaces, and PortChannel subinterfaces
iSCSI hosts mapping to virtual Fibre Channel hosts	Dynamic mapping.

*Table 22-4 Default iSCSI Parameters (continued)*

Parameters	Default
Dynamic iSCSI initiators	Members of the VSAN 1.
Identifying initiators	iSCSI node names.
Advertising static virtual targets	No initiators are allowed to access a virtual target (unless explicitly configured).
iSCSI login authentication	CHAP or none authentication mechanism.
revert-primary-port	Disabled.
Header and data digest	Enabled automatically when iSCSI initiators send requests. This feature cannot be configured and is not available in store-and-forward mode.

