Send documentation comments to mdsfeedback-doc@cisco.com

Cisco MDS 9000 Family NX-OS System Management Configuration Guide

Cisco MDS NX-OS Release 5.0(1a)
February 2010

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-21480-01

Cisco MDS 9000 Family NX-OS System Management Configuration Guide

CONTENTS

New and Changed Information

As of Cisco MDS NX-OS Release 4.2(1), software configuration information is available in new feature-specific configuration guides for the following information:

- System management
- Interfaces
- Fabric
- Quality of service
- Security
- IP services
- High availability and redundancy

The information in these new guides previously existed in the *Cisco MDS 9000 Family CLI Configuration Guide* and in the *Cisco MDS 9000 Family Fabric Manager Configuration Guide*. Those configuration guides remain available on Cisco.com and should be used for all software releases prior to MDS NX-OS Release 4.2(1). Each guide addresses the features introduced in or available in a particular release. Select and view the configuration guide that pertains to the software installed in your switch.

For a complete list of document titles, see the list of Related Documentation in the “Preface.”

To find additional information about Cisco MDS NX-OS Release 4.2(x), see the *Cisco MDS 9000 Family Release Notes* available at the following Cisco Systems website:

http://www.cisco.com/en/US/products/ps5989/prod_release_notes_list.htm

About this Guide

The information in the new *Cisco MDS 9000 NX-OS System Management Configuration Guide* previously existed in the following parts of the *Cisco MDS 9000 Family CLI Configuration Guide:*

- Part 2: Installation and Switch Management
- Part 5: Security
- Part 8: Network and Switch Monitoring
- Part 9: Troubleshooting

Table 1 lists the New and Changed features for this guide, starting with MDS NX-OS Release 4.2(1).

Table i-1 New and Changed Features for Cisco MDS NX-OS Release 5.0(1a)

Feature	New or Changed Topics	Changed in Release	Where Documented
Multiple SMTP Server Support	Added Multiple SMTP Server Support details. Added Verifying Callhome Transport Commands.	5.0(1a)	Chapter 4, “Configuring Call Home”
Call Home Destination tab	Added the enhancement in Destination tab.	4.2(1)	Chapter 4, “Configuring Call Home”
Call Home HTTPs support	Added Call Home HTTPs enhancement.	4.2(1)	Chapter 4, “Configuring Call Home”
SNMP Trap Control tab	Added details of the new Control tab available from NX-OS Release 4.2(1).	4.2(1)	Chapter 9, “Configuring SNMP”
Domain Manager Turbo Mode	Added procedure to configure Domain Manager turbo mode.	4.2(1)	Chapter 10, “Configuring Domain Parameters”

As of Cisco MDS NX-OS Release 4.2(1), software configuration information is available in new feature-specific configuration guides for the following information:

- System management
- Interfaces
- Fabric
- Quality of service
- Security
- IP services
- High availability and redundancy

The information in these new guides previously existed in the *Cisco MDS 9000 Family CLI Configuration Guide* and in the *Cisco MDS 9000 Family Fabric Manager Configuration Guide*. Those configuration guides remain available on Cisco.com and should be used for all software releases prior to MDS NX-OS Release 4.2(1). Each guide addresses the features introduced in or available in a particular release. Select and view the configuration guide that pertains to the software installed in your switch.

Some information from the *Cisco MDS 9000 Family CLI Configuration Guide* and the *Cisco MDS 9000 Family Fabric Manager Configuration Guide* now appears in the following guides that are common among products that run the Nexus operating system:

- *Cisco NX-OS Family Licensing Guide* – Explains the licensing model and describes the feature licenses.
- *Cisco NX-OS Fundamentals Configuration Guide* – Describes the switch setup utility and includes general CLI, file system, and configuration information.

For a complete list of document titles, see the list of Related Documentation in the “Preface.”

To find additional information about Cisco MDS NX-OS Release 4.2(x), see the *Cisco MDS 9000 Family Release Notes* available at the following Cisco Systems website:

http://www.cisco.com/en/US/products/ps5989/prod_release_notes_list.htm

About this Guide

The information in the new *Cisco Fabric Manager System Management Configuration Guide* previously existed in the following parts of the *Cisco MDS 9000 Family Fabric Manager Configuration Guide:*

- Part 2: Installation and Switch Management
- Part 5: Security
- Part 8: Network and Switch Monitoring
- Part 9: Troubleshooting

Preface

This preface describes the audience, organization, and conventions of the *Cisco MDS 9000 Family NX-OS System Management Configuration Guide*. It also provides information on how to obtain related documentation.

Audience

This guide is for experienced network administrators who are responsible for configuring and maintaining the Cisco MDS 9000 Family of multilayer directors and fabric switches.

Organization

This guide is organized as follows:

Chapter	Title	Description
Chapter 1	System Management Overview	Provides an overview of the system management features to monitor and manage a switch using the CLI.
Chapter 2	Using the CFS Infrastructure	Explains the use of the Cisco Fabric Services (CFS) infrastructure to enable efficient database distribution.
Chapter 3	Configuring System Message Logging	Describes how system message logging is configured and displayed.
Chapter 4	Configuring Call Home	Provides details on the Call Home service and includes information on Call Home, event triggers, contact information, destination profiles, and e-mail options.
Chapter 5	Scheduling Maintenance Jobs	Describes the Cisco MDS command scheduler feature that helps you schedule configuration and maintenance jobs in any switch in the Cisco MDS 9000 Family.

Chapter	Title	Description
Chapter 6	Monitoring System Processes and Logs	Provides information on displaying system processes and status. It also provides information on configuring core and log files, HA policy, heartbeat and watchdog checks, and upgrade resets.
Chapter 7	Configuring the Embedded Event Manager	Provides information about configuring Embedded Event Manager (EEM).
Chapter 9	Configuring SNMP	Provides details on how you can use SNMP to modify a role that was created using CLI.
Chapter 8	Configuring RMON	Provides details on using RMONs to configure alarms and events.
Chapter 10	Configuring Domain Parameters	Explains the Fibre Channel domain (fcdomain) feature, which includes principal switch selection, domain ID distribution, FC ID allocation, and fabric reconfiguration functions.
Chapter 11	Monitoring Network Traffic Using SPAN	Describes the Switched Port Analyzer (SPAN), SPAN sources, filters, SPAN sessions, SD port characteristics, and configuration details.
Chapter 12	Configuring Fabric Configuration Server	Describes how the fabric configuration server (FCS) feature is configured and displayed.

Document Conventions

Command descriptions use these conventions:

boldface font	Commands and keywords are in boldface.
italic font	Arguments for which you supply values are in italics.
[ ]	Elements in square brackets are optional.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Screen examples use these conventions:

`screen font`	Terminal sessions and information the switch displays are in screen font.
`boldface screen font`	Information you must enter is in boldface screen font.
italic screen font	Arguments for which you supply values are in italic screen font.
`< >`	Nonprinting characters, such as passwords, are in angle brackets.
`[ ]`	Default responses to system prompts are in square brackets.
`!, #`	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

This document uses the following conventions:

Send documentation comments to mdsfeedback-doc@cisco.com

Note Means reader *take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Related Documentation

The documentation set for the Cisco MDS 9000 Family includes the following documents. To find a document online, use the Cisco MDS NX-OS Documentation Locator at:

http://www.cisco.com/en/US/docs/storage/san_switches/mds9000/roadmaps/doclocater.htm

Release Notes

- *Cisco MDS 9000 Family Release Notes for Cisco MDS NX-OS Releases*
- *Cisco MDS 9000 Family Release Notes for MDS SAN-OS Releases*
- *Cisco MDS 9000 Family Release Notes for Storage Services Interface Images*
- *Cisco MDS 9000 Family Release Notes for Cisco MDS 9000 EPLD Images*
- *Release Notes for Cisco MDS 9000 Family Fabric Manager*

Regulatory Compliance and Safety Information

- *Regulatory Compliance and Safety Information for the Cisco MDS 9000 Family*

Compatibility Information

- *Cisco Data Center Interoperability Support Matrix*
- *Cisco MDS 9000 NX-OS Hardware and Software Compatibility Information and Feature Lists*
- *Cisco MDS NX-OS Release Compatibility Matrix for Storage Service Interface Images*
- *Cisco MDS 9000 Family Switch-to-Switch Interoperability Configuration Guide*
- *Cisco MDS NX-OS Release Compatibility Matrix for IBM SAN Volume Controller Software for Cisco MDS 9000*
- *Cisco MDS SAN-OS Release Compatibility Matrix for VERITAS Storage Foundation for Networks Software*

Hardware Installation

- *Cisco MDS 9500 Series Hardware Installation Guide*

- *Cisco MDS 9200 Series Hardware Installation Guide*
- *Cisco MDS 9100 Series Hardware Installation Guide*
- *Cisco MDS 9124 and Cisco MDS 9134 Multilayer Fabric Switch Quick Start Guide*

Software Installation and Upgrade

- *Cisco MDS 9000 NX-OS Release 4.1(x) and SAN-OS 3(x) Software Upgrade and Downgrade Guide*
- *Cisco MDS 9000 Family Storage Services Interface Image Install and Upgrade Guide*
- *Cisco MDS 9000 Family Storage Services Module Software Installation and Upgrade Guide*

Cisco NX-OS

- *Cisco MDS 9000 Family NX-OS Licensing Guide*
- *Cisco MDS 9000 Family NX-OS Fundamentals Configuration Guide*
- *Cisco MDS 9000 Family NX-OS Interfaces Configuration Guide*
- *Cisco MDS 9000 Family NX-OS Fabric Configuration Guide*
- *Cisco MDS 9000 Family NX-OS Quality of Service Configuration Guide*
- *Cisco MDS 9000 Family NX-OS Security Configuration Guide*
- *Cisco MDS 9000 Family NX-OS IP Services Configuration Guide*
- *Cisco MDS 9000 Family NX-OS Intelligent Storage Services Configuration Guide*
- *Cisco MDS 9000 Family NX-OS High Availability and Redundancy Configuration Guide*
- *Cisco MDS 9000 Family NX-OS Inter-VSAN Routing Configuration Guide*

Cisco Fabric Manager

- *Cisco Fabric Manager Fundamentals Configuration Guide*
- *Cisco Fabric Manager System Management Configuration Guide*
- *Cisco Fabric Manager Interfaces Configuration Guide*
- *Cisco Fabric Manager Fabric Configuration Guide*
- *Cisco Fabric Manager Quality of Service Configuration Guide*
- *Cisco Fabric Manager Security Configuration Guide*
- *Cisco Fabric Manager IP Services Configuration Guide*
- *Cisco Fabric Manager Intelligent Storage Services Configuration Guide*
- *Cisco Fabric Manager High Availability and Redundancy Configuration Guide*
- *Cisco Fabric Manager Inter-VSAN Routing Configuration Guide*
- *Cisco Fabric Manager Online Help*
- *Cisco Fabric Manager Web Services Online Help*

Send documentation comments to mdsfeedback-doc@cisco.com

Command-Line Interface

- *Cisco MDS 9000 Family Command Reference*

Intelligent Storage Networking Services Configuration Guides

- *Cisco MDS 9000 I/O Acceleration Configuration Guide*
- *Cisco MDS 9000 Family SANTap Deployment Guide*
- *Cisco MDS 9000 Family Data Mobility Manager Configuration Guide*
- *Cisco MDS 9000 Family Storage Media Encryption Configuration Guide*
- *Cisco MDS 9000 Family Secure Erase Configuration Guide*
- *Cisco MDS 9000 Family Cookbook for Cisco MDS SAN-OS*

Troubleshooting and Reference

- *Cisco NX-OS System Messages Reference*
- *Cisco MDS 9000 Family NX-OS Troubleshooting Guide*
- *Cisco MDS 9000 Family NX-OS MIB Quick Reference*
- *Cisco MDS 9000 Family NX-OS SMI-S Programming Reference*
- *Cisco MDS 9000 Family Fabric Manager Server Database Schema*

CHAPTER 1

System Management Overview

You can use the system management features to monitor and manage a switch using the Fabric Manager. These features include Call Home, SNMP, RMON, SPAN, and the Embedded Event Manager (EEM).

This chapter describes these features and includes the following sections:

Cisco Fabric Services

The Cisco MDS NX-OS software uses the Cisco Fabric Services (CFS) infrastructure to enable efficient database distribution and to promote device flexibility. CFS simplifies SAN provisioning by automatically distributing configuration information to all switches in a fabric.

For information on configuring CFS, see Chapter 2, “Using the CFS Infrastructure.”

System Messages

System messages are monitored remotely by accessing the switch through Telnet, SSH, or the console port, or by viewing the logs on a system message logging server. Log messages are not saved across system reboots.

For information about configuring system messages, see Chapter 3, “Configuring System Message Logging.”

Call Home

Call Home provides e-mail-based notification of critical system events. A versatile range of message formats are available for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications. Common uses of this feature may include direct paging of a network support engineer, e-mail notification to a Network Operations Center, and utilization of Cisco Smart Call Home services for direct case generation with the Technical Assistance Center.

For information about configuring Call Home, see Chapter 4, “Configuring Call Home.”

Scheduler

The Cisco MDS command scheduler feature helps you schedule configuration and maintenance jobs in any switch in the Cisco MDS 9000 Family switches.You can use this feature to schedule jobs on a one-time basis or periodically. The Cisco NX-OS command scheduler provides a facility to schedule a job (set of CLI commands) or multiple jobs at a specified time in the future. The jobs can be executed once at a specified time in the future or at periodic intervals.

For information on configuring the Cisco MDS command scheduler feature, see Chapter 5, “Scheduling Maintenance Jobs.”

System Processes and Logs

The health of a switch can be monitored by various system processes and logs. The Online Health Management System (system health) is a hardware fault detection and recovery feature. This Health Management System ensures the general health of switching, services, and supervisor modules in any switch in the Cisco MDS 9000 Family.

For information on monitoring the health of the switch, see Chapter 6, “Monitoring System Processes and Logs.”

Embedded Event Manager

Embedded Event Manager (EEM) monitors events that occur on your device and takes action to recover or troubleshoot these events, based on your configuration. EEM consists of three major components:

- Event statements—Events to monitor from another Cisco NX-OS component that may require some action, workaround, or notification.
- Action statements —An action that EEM can take, such as sending an e-mail or disabling an interface, to recover from an event.
- Policies—An event paired with one or more actions to troubleshoot or recover from the event.

For information on configuring EEM, see Chapter 7, “Configuring the Embedded Event Manager.”

SNMP

Simple Network Management Protocol (SNMP) is an application layer protocol that facilitates the exchange of management information between network devices. In all Cisco MDS 9000 Family switches, three SNMP versions are available: SNMPv1, SNMPv2c, and SNMPv3. The CLI and SNMP use common roles in all switches in the Cisco MDS 9000 Family. You can use SNMP to modify a role that was created using the CLI and vice versa.

Users, passwords, and roles for all CLI and SNMP users are the same. A user configured through the CLI can access the switch using SNMP (for example, the Fabric Manager or the Device Manager) and vice versa.

For information on configuring SNMP, see Chapter 7, “Configuring SNMP.”

RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. You can use the RMON alarms and events to monitor Cisco MDS 9000 Family switches running the Cisco SAN-OS Release 2.0(1b) or later or Cisco Release NX-OS 4.1(3) or later software.

For information on configuring RMON, see Chapter 8, “Configuring RMON.”

Domain Parameters

The Fibre Channel domain (fcdomain) feature performs principal switch selection, domain ID distribution, FC ID allocation, and fabric reconfiguration functions as described in the FC-SW-2 standards. The domains are configured on a per-VSAN basis. If you do not configure a domain ID, the local switch uses a random ID.

For information on configuring the Fibre Channel domain feature, see Chapter 9, “Configuring Domain Parameters.”

SPAN

The Switched Port Analyzer (SPAN) feature is specific to switches in the Cisco MDS 9000 Family. It monitors network traffic through a Fibre Channel interface. Traffic through any Fibre Channel interface can be replicated to a special port called the SPAN destination port (SD port). Any Fibre Channel port in a switch can be configured as an SD port. Once an interface is in SD port mode, it cannot be used for normal data traffic. You can attach a Fibre Channel analyzer to the SD port to monitor SPAN traffic.

For information on SPAN feature, see Chapter 10, “Monitoring Network Traffic Using SPAN.”

Fabric Configuration Server

The Fabric Configuration Server (FCS) provides discovery of topology attributes and maintains a repository of configuration information of fabric elements. A management application is usually connected to the FCS on the switch through an N port. In the Cisco MDS 9000 Family switch environment, multiple VSANs constitute a fabric, where one instance of the FCS is present per VSAN.

For information on configuring FCS, see Chapter 11, “Configuring Fabric Configuration Servers.”

CHAPTER 2

Using the CFS Infrastructure

This chapter contains the following sections:

About CFS

The Cisco MDS NX-OS software uses the Cisco Fabric Services (CFS) infrastructure to enable efficient database distribution and to foster device flexibility. It simplifies SAN provisioning by automatically distributing configuration information to all switches in a fabric.

Several Cisco MDS NX-OS applications use the CFS infrastructure to maintain and distribute the contents of a particular application's database.

Many features in the Cisco MDS switches require configuration synchronization in all switches in the fabric. Maintaining configuration synchronization across a fabric is important to maintain fabric consistency. In the absence of a common infrastructure, such synchronization is achieved through manual configuration at each switch in the fabric. This process is tedious and error prone.

Cisco Fabric Services (CFS) provides a common infrastructure for automatic configuration synchronization in the fabric. It provides the transport function as well as a rich set of common services to the applications. CFS has the ability to discover CFS capable switches in the fabric and discovering application capabilities in all CFS capable switches.

This section includes the following topics:

Cisco MDS NX-OS Features Using CFS

The following Cisco NX-OS features use the CFS infrastructure:

- N Port Virtualization
- FlexAttach Virtual pWWN
- NTP
- Dynamic Port VSAN Membership
- Distributed Device Alias Services
- IVR topology
- SAN device virtualization
- TACACS+ and RADIUS
- User and administrator roles
- Port security
- iSNS
- Call Home
- Syslog
- fctimer
- SCSI flow services
- Saved startup configurations using the Fabric Startup Configuration Manager (FSCM)
- Allowed domain ID lists
- RSCN timer
- iSLB

CFS Features

CFS has the following features:

- Peer-to-peer protocol with no client-server relationship at the CFS layer.
- Three scopes of distribution.
  - Logical scope—The distribution occurs within the scope of a VSAN.
  - Physical scope—The distribution spans the entire physical topology.
  - Over a selected set of VSANs—Some applications, such as Inter-VSAN Routing (IVR), require configuration distribution over some specific VSANs. These applications can specify to CFS the set of VSANs over which to restrict the distribution.

- Three modes of distribution.
  - Coordinated distributions—Only one distribution is allowed in the fabric at any given time.
  - Uncoordinated distributions—Multiple parallel distributions are allowed in the fabric except when a coordinated distribution is in progress.
  - Unrestricted uncoordinated distributions—Multiple parallel distributions are allowed in the fabric in the presence of an existing coordinated distribution. Unrestricted uncoordinated distributions are allowed to run in parallel with all other types of distributions.
- Supports a merge protocol that facilitates the merge of application configuration during a fabric merge event (when two independent fabrics merge).

CFS Protocol

The CFS functionality is independent of the lower layer transport. Currently, in Cisco MDS switches, the CFS protocol layer resides on top of the Fiber Channel 22 (FC2) layer and is peer-to-peer with no client-server relationship. CFS uses the FC2 transport services to send information to other switches. CFS uses a proprietary SW_ILS (0x77434653) protocol for all CFS packets. CFS packets are sent to or from the switch domain controller addresses.

CFS can also use IP to send information to other switches.

Applications that use CFS are completely unaware of the lower layer transport.

CFS Distribution Scopes

Different applications on the Cisco MDS 9000 Family switches need to distribute the configuration at various levels:

- VSAN level (logical scope)

  Applications that operate within the scope of a VSAN have the configuration distribution restricted to the VSAN. An example application is port security where the configuration database is applicable only within a VSAN.

- Physical topology level (physical scope)

  Applications might need to distribute the configuration to the entire physical topology spanning several VSANs. Such applications include NTP and DPVM (WWN-based VSAN), which are independent of VSANs.

- Between selected switches

  Applications might only operate between selected switches in the fabric. An example application is SCSI flow services, which operates between two switches.

CFS Distribution Modes

CFS supports different distribution modes to support different application requirements: coordinated and uncoordinated distributions. Both modes are mutually exclusive. Only one mode is allowed at any given time.

Uncoordinated Distribution

Uncoordinated distributions are used to distribute information that is not expected to conflict with that from a peer. An example is local device registrations such as iSNS. Parallel uncoordinated distributions are allowed for an application.

Coordinated Distribution

Coordinated distributions can have only one application distribution at a given time. CFS uses locks to enforce this. A coordinated distribution is not allowed to start if locks are taken for the application anywhere in the fabric. A coordinated distribution consists of three stages:

1. A fabric lock is acquired.
2. The configuration is distributed and committed.
3. The fabric lock is released.

Coordinated distribution has two variants:

- CFS driven —The stages are executed by CFS in response to an application request without intervention from the application.
- Application driven—The stages are under the complete control of the application.

  Coordinated distributions are used to distribute information that can be manipulated and distributed from multiple switches, for example, the port security configuration.

Unrestricted Uncoordinated Distributions

Unrestricted uncoordinated distributions allow multiple parallel distributions in the fabric in the presence of an existing coordinated distribution. Unrestricted uncoordinated distributions are allowed to run in parallel with all other types of distributions.

Disabling CFS Distribution on a Switch

By default, CFS distribution is enabled. Applications can distribute data and configuration information to all CFS-capable switches in the fabric where the applications exist. This is the normal mode of operation.

You can globally disable CFS on a switch, including CFS over IP to isolate the applications using CFS from fabric-wide distributions while maintaining physical connectivity. When CFS is globally disabled on a switch, CFS operations are restricted to the switch and all CFS commands continue to function as if the switch were physically isolated.

To globally disable or enable CFS distribution on a switch, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# no cfs distribute`	Globally disables CFS distribution for all applications on the switch, including CFS over IP.
	`switch(config)# cfs distribute`	Enables (default) CFS distribution on the switch.

Verifying CFS Distribution Status

The **show cfs status** command displays the status of CFS distribution on the switch.

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Disabled
IPv4 multicast address : 239.255.70.83
IPv6 multicast address : ff15::efff:4653
```

CFS Application Requirements

All switches in the fabric must be CFS capable. A Cisco MDS 9000 Family switch is CFS capable if it is running Cisco SAN-OS Release 2.0(1b) or later, or MDS NX-OS Release 4.1(1) or later. Switches that are not CFS capable do not receive distributions and result in part of the fabric not receiving the intended distribution.

CFS has the following requirements:

- Implicit CFS usage—The first time you issue a CFS task for a CFS-enabled application, the configuration modification process begins and the application locks the fabric.
- Pending database—The pending database is a temporary buffer to hold uncommitted information. The uncommitted changes are not applied immediately to ensure that the database is synchronized with the database in the other switches in the fabric. When you commit the changes, the pending database overwrites the configuration database (also known as the active database or the effective database).
- CFS distribution enabled or disabled on a per-application basis—The default (enable or disable) for CFS distribution state differs between applications. If CFS distribution is disabled for an application, then that application does not distribute any configuration nor does it accept a distribution from other switches in the fabric.
- Explicit CFS commit—Most applications require an explicit commit operation to copy the changes in the temporary buffer to the application database, to distribute the new database to the fabric, and to release the fabric lock. The changes in the temporary buffer are not applied if you do not perform the commit operation.

Enabling CFS for an Application

All CFS-based applications provide an option to enable or disable the distribution capabilities. Features that existed prior to Cisco SAN-OS Release 2.0(1b) have the distribution capability disabled by default and must have distribution capabilities enabled explicitly.

Applications introduced in Cisco SAN-OS Release 2.0(1b) or later, or MDS NX-OS Release 4.1(1) or later have the distribution enabled by default.

The application configuration is not distributed by CFS unless distribution is explicitly enabled for that application.

Verifying Application Registration Status

The **show cfs application** command displays the applications that are currently registered with CFS. The first column displays the application name. The second column indicates whether the application is enabled or disabled for distribution (enabled or disabled). The last column indicates the scope of distribution for the application (logical, physical, or both).

Note The **show cfs application** command only displays applications registered with CFS. Conditional services that use CFS do not appear in the output unless these services are running.

```
switch# show cfs application
----------------------------------------------
 Application    Enabled   Scope
----------------------------------------------
 ntp            No        Physical-all
 fscm           Yes       Physical-fc
 islb           No        Physical-fc
 role           No        Physical-all
 rscn           No        Logical
 radius         No        Physical-all
 fctimer        No        Physical-fc
 syslogd        No        Physical-all
 callhome       No        Physical-all
 fcdomain       No        Logical
 device-alias   Yes       Physical-fc

Total number of entries = 11
```

The **show cfs application name** command displays the details for a particular application. It displays the enabled/disabled state, timeout as registered with CFS, merge capability (if it has registered with CFS for merge support), and the distribution scope.

```
switch# show cfs application name ntp

 Enabled        : Yes
 Timeout        : 5s
 Merge Capable  : Yes
 Scope          : Physical
 Region        : Default
```

Locking the Fabric

When you configure (first time configuration) a Cisco NX-OS feature (or application) that uses the CFS infrastructure, that feature starts a CFS session and locks the fabric. When a fabric is locked, the Cisco NX-OS software does not allow any configuration changes from a switch to this Cisco NX-OS feature, other than the switch holding the lock, and issues a message to inform the user about the locked status. The configuration changes are held in a pending database by that application.

If you start a CFS session that requires a fabric lock but forget to end the session, an administrator can clear the session. If you lock a fabric at any time, your user name is remembered across restarts and switchovers. If another user (on the same machine) tries to perform configuration tasks, that user's attempts are rejected.

Verifying CFS Lock Status

The **show cfs lock** command displays all the locks that are currently acquired by any application. For each application the command displays the application name and scope of the lock taken. If the application lock is taken in the physical scope, then this command displays the switch WWN, IP address, user name, and user type of the lock holder. If the application is taken in the logical scope, then this command displays the VSAN in which the lock is taken, the domain, IP address, user name, and user type of the lock holder.

```
switch# show cfs lock

Application: ntp
Scope      : Physical
--------------------------------------------------------------------
 Switch WWN               IP Address       User Name    User Type
--------------------------------------------------------------------
 20:00:00:05:30:00:6b:9e  10.76.100.167    admin        CLI/SNMP v3
 Total number of entries = 1

Application: port-security
Scope      : Logical
----------------------------------------------------------
 VSAN   Domain   IP Address       User Name    User Type
----------------------------------------------------------
 1      238      10.76.100.167    admin        CLI/SNMP v3
 2      211      10.76.100.167    admin        CLI/SNMP v3
 Total number of entries = 2
```

The **show cfs lock name** command displays the lock details similar for the specified application.

Example 2-1 Displays the Lock Information for the Specified Application

```
switch# show cfs lock name ntp
Scope      : Physical
--------------------------------------------------------------------
 Switch WWN               IP Address       User Name    User Type
--------------------------------------------------------------------
 20:00:00:05:30:00:6b:9e  10.76.100.167    admin        CLI/SNMP v3

 Total number of entries = 1
```

Committing Changes

A commit operation saves the pending database for all application peers and releases the lock for all switches.

In general, the commit function does not start a session; only a lock function starts a session. However, an empty commit is allowed if configuration changes are not previously made. In this case, a commit operation results in a session that acquires locks and distributes the current database.

When you commit configuration changes to a feature using the CFS infrastructure, you receive a notification about one of the following responses:

- One or more external switches report a successful status—The application applies the changes locally and releases the fabric lock.
- None of the external switches report a successful state—The application considers this state a failure and does not apply the changes to any switch in the fabric. The fabric lock is not released.

You can commit changes for a specified feature by entering the **commit** command for that feature.

Discarding Changes

If you discard configuration changes, the application flushes the pending database and releases locks in the fabric. Both the abort and commit functions are only supported from the switch from which the fabric lock is acquired.

You can discard changes for a specified feature by using the **abort** command for that feature.

Saving the Configuration

Configuration changes that have not been applied yet (still in the pending database) are not shown in the running configuration. The configuration changes in the pending database overwrite the configuration in the effective database when you commit the changes.

Caution If you do not commit the changes, they are not saved to the running configuration.

The CISCO-CFS-MIB contains SNMP configuration information for any CFS-related functions. Refer to the *Cisco MDS 9000 Family MIB Quick Reference* for more information on this MIB.

Clearing a Locked Session

You can clear locks held by an application from any switch in the fabric. This option is provided to rescue you from situations where locks are acquired and not released. This function requires Admin permissions.

The CFS locks can be cleared by using the following methods in the CLI:

- Abort the configuration from the switch where the configuration lock was acquired previously. This method clears the CFS locks in the entire fabric.

  This example shows how to clear the CFS locks for the DPVM application in the entire fabric:

```
Switch# Conf t
Switch(conf)# dpvm abort
```

- Clear the sessions from any switch in the fabric.

  This example shows how to clear the CFS locks for the DPVM application:

```
Switch# Conf t
Switch# clear dpvm session
```

Caution Exercise caution when using this function to clear locks in the fabric. Any pending configurations in any switch in the fabric is flushed and lost.

CFS Merge Support

An application keeps the configuration synchronized in a fabric through CFS. Two such fabrics might merge as a result of an ISL coming up between them. These two fabrics could have two different sets of configuration information that need to be reconciled in the event of a merge. CFS provides notification each time an application peer comes online. If a fabric with M application peers merges with another fabric with N application peers and if an application triggers a merge action on every such notification, a link-up event results in M*N merges in the fabric.

CFS supports a protocol that reduces the number of merges required to one by handling the complexity of the merge at the CFS layer. This protocol runs per application per scope. The protocol involves selecting one switch in a fabric as the merge manager for that fabric. The other switches do not play any role in the merge process.

During a merge, the merge manager in the two fabrics exchange their configuration databases with each other. The application on one of them merges the information, decides if the merge is successful, and informs all switches in the combined fabric of the status of the merge.

In case of a successful merge, the merged database is distributed to all switches in the combined fabric and the entire new fabric remains in a consistent state. You can recover from a merge failure by starting a distribution from any of the switches in the new fabric. This distribution restores all peers in the fabric to the same configuration database.

Verifying CFS Merge Status

The **show cfs merge status name** command displays the merge status for a given application. The following example displays the output for an application distributing in logical scope. It shows the merge status in all valid VSANs on the switch. The command output shows the merge status as one of the following: Success, Waiting, Failure, or In Progress. In case of a successful merge, all the switches in the fabric are shown under the local fabric. In case of a merge failure or a merge in progress, the local fabric and the remote fabric involved in the merge are indicated separately. The application server in each fabric that is mainly responsible for the merge is indicated by the term Merge Master.

```
switch# show cfs merge status name port-security

Logical [VSAN 1] Merge Status: Failed
Local Fabric
-----------------------------------------------------------------
 Domain Switch WWN               IP Address
-----------------------------------------------------------------
 238    20:00:00:05:30:00:6b:9e  10.76.100.167   [Merge Master]

Remote Fabric
-----------------------------------------------------------------
 Domain Switch WWN               IP Address
-----------------------------------------------------------------
 236    20:00:00:0e:d7:00:3c:9e  10.76.100.169    [Merge Master]

Logical [VSAN 2] Merge Status: Success
Local Fabric
-----------------------------------------------------------------
 Domain Switch WWN               IP Address
-----------------------------------------------------------------
 211    20:00:00:05:30:00:6b:9e  10.76.100.167    [Merge Master]
 1      20:00:00:0e:d7:00:3c:9e  10.76.100.169

Logical [VSAN 3] Merge Status: Success
Local Fabric
```

```
-----------------------------------------------------------------
 Domain Switch WWN               IP Address
-----------------------------------------------------------------
 221    20:00:00:05:30:00:6b:9e  10.76.100.167    [Merge Master]
 103    20:00:00:0e:d7:00:3c:9e  10.76.100.169
```

The following example of the **show cfs merge status name** command output displays an application using the physical scope with a merge failure. The command uses the specified application name to display the merge status based on the application scope.

```
switch# show cfs merge status name ntp

Physical  Merge Status: Failed
Local Fabric
-----------------------------------------------------------
 Switch WWN               IP Address
-----------------------------------------------------------
 20:00:00:05:30:00:6b:9e  10.76.100.167    [Merge Master]

Remote Fabric
-----------------------------------------------------------
 Switch WWN               IP Address
-----------------------------------------------------------
 20:00:00:0e:d7:00:3c:9e  10.76.100.169    [Merge Master]
```

The **show cfs peers** command output displays all the switches in the physical fabric in terms of the switch WWN and the IP address. The local switch is indicated as Local.

```
switch# show cfs peers

Physical Fabric
-------------------------------------------------
 Switch WWN               IP Address
-------------------------------------------------
 20:00:00:05:30:00:6b:9e  10.76.100.167   [Local]
 20:00:00:0e:d7:00:3c:9e  10.76.100.169

 Total number of entries = 2
```

The **show cfs peers name** command displays all the peers for which a particular application is registered with CFS. The command output shows all the peers for the physical scope or for each of the valid VSANs on the switch, depending on the application scope. For physical scope, the switch WWNs for all the peers are indicated. The local switch is indicated as Local.

```
switch# show cfs peers name ntp

Scope      : Physical
-------------------------------------------------
Switch WWN               IP Address
-------------------------------------------------
20:00:00:44:22:00:4a:9e  172.22.92.27    [Local]
20:00:00:05:30:01:1b:c2  172.22.92.215
```

The following example **show cfs peers name** command output displays all the application peers (all switches in which that application is registered). The local switch is indicated as Local.

```
switch# show cfs peers name port-security
Scope      : Logical [VSAN 1]
-----------------------------------------------------------
Domain   Switch WWN               IP Address
-----------------------------------------------------------
124      20:00:00:44:22:00:4a:9e  172.22.92.27    [Local]
```

```
98       20:00:00:05:30:01:1b:c2  172.22.92.215

Total number of entries = 2


Scope      : Logical [VSAN 3]
--------------------------------------------------------
Domain   Switch WWN               IP Address
--------------------------------------------------------
224      20:00:00:44:22:00:4a:9e  172.22.92.27    [Local]
151      20:00:00:05:30:01:1b:c2  172.22.92.215

Total number of entries = 2
```

CFS Distribution over IP

You can configure CFS to distribute information over IP for networks containing switches that are not reachable over Fibre Channel. CFS distribution over IP supports the following features:

- Physical distribution over an entirely IP network.
- Physical distribution over a hybrid Fibre Channel and IP network with the distribution reaching all switches that are reachable over either Fibre Channel or IP.

Note The switch attempts to distribute information over Fibre Channel first and then over the IP network if the first attempt over Fibre Channel fails. CFS does not send duplicate messages if distribution over both IP and Fibre Channel is enabled.

- Distribution over IP version 4 (IPv4) or IP version 6 (IPv6).

Note CFS cannot distribute over both IPv4 and IPv6 from the same switch.

- Keepalive mechanism to detect network topology changes using a configurable multicast address.
- Compatibility with Cisco MDS SAN-OS Release 2.x.
- Distribution for logical scope applications is not supported because the VSAN implementation is limited to Fibre Channel.

Figure 2-1 shows a network with both Fibre Channel and IP connections. Node A forwards an event to node B over Fibre Channel. Node B forwards the event node C and node D using unicast IP. Node C forwards the event to node E using Fibre Channel.

Figure 2-1 Network Example 1 with Fibre Channel and IP Connections

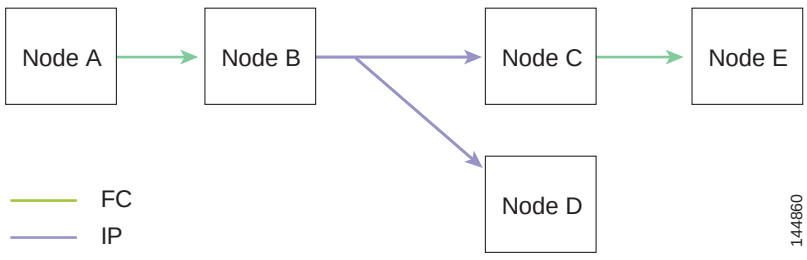


Figure 2-2 is the same as Figure 2-1 except that node D and node E are connected using Fibre Channel. All processes is the same in this example because node B has node C and node D the distribution list for IP. Node C does not forward to node D because node D is already in the distribution list from node B.

Figure 2-2 Network Example 2 with Fibre Channel and IP Connections

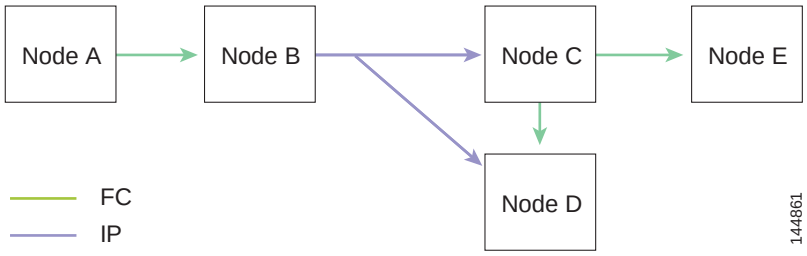
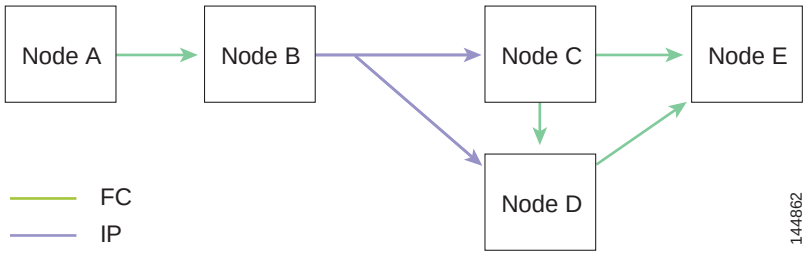


Figure 2-3 is the same as Figure 2-2 except that node D and node E are connected using IP. Both node C and node D forward the event to E because the node E is not in the distribution list from node B.

Figure 2-3 Network Example 3 with Fibre Channel and IP Connections



Enabling CFS over IP

To enable or disable CFS over IPv4, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# cfs ipv4 distribute`	Globally enables CFS over IPv4 for all applications on the switch.

	`switch(config)# no cfs ipv4 distribute`
	`This will prevent CFS from distributing over IPv4 network.`
	`Are you sure? (y/n) [n] y`
		Disables (default) CFS over IPv4 on the switch.

To enable or disable CFS over IPv6, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.

	Command	Purpose
Step 2	`switch(config)# cfs ipv6 distribute`	Globally enables CFS over IPv6 for all applications on the switch.
	`switch(config)# no cfs ipv6 distribute`	Disables (default) CFS over IPv6 on the switch.

Verifying the CFS over IP Configuration

To verify the CFS over IP configuration, use the **show cfs status** command.

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Disabled
IPv4 multicast address : 239.255.70.83
IPv6 multicast address : ff15::efff:4653
```

Configuring IP Multicast Address for CFS over IP

All CFS over IP enabled switches with similar multicast addresses form one CFS over IP fabric. CFS protocol specific distributions, such as the keepalive mechanism for detecting network topology changes, use the IP multicast address to send and receive information.

Note CFS distributions for application data use directed unicast.

You can configure a CFS over IP multicast address value for either IPv4 or IPv6. The default IPv4 multicast address is 239.255.70.83 and the default IPv6 multicast address is ff15:efff:4653.

To configure an IP multicast address for CFS over IPv4, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.

Step 2	`switch(config)# cfs ipv4 mcast-address 239.255.1.1`
	`Distribution over this IP type will be affected`
	`Change multicast address for CFS-IP ?`
	`Are you sure? (y/n) [n] y`
		Configures the IPv4 multicast address for CFS distribution over IPv4. The ranges of valid IPv4 addresses are 239.255.0.0 through 239.255.255.255 and 239.192/16 through 239.251/16.

	`switch(config)# no cfs ipv4 mcast-address 239.255.1.1`
	`Distribution over this IP type will be affected`
	`Change multicast address for CFS-IP ?`
	`Are you sure? (y/n) [n] y`
		Reverts to the default IPv4 multicast address for CFS distribution over IPv4. The default IPv4 multicast address for CFS is 239.255.70.83.

To configure an IP multicast address for CFS over IPv6, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.

Step 2	`switch(config)# cfs ipv6 mcast-address ff15::e244:4754`
	`Distribution over this IP type will be affected`
	`Change multicast address for CFS-IP ?`
	`Are you sure? (y/n) [n] y`
		Configures the IPv6 multicast address for CFS distribution over IPv6. The range of valid IPv6 addresses is ff15::/16 (ff15::0000:0000 through ff15::ffff:ffff) and ff18::/16 (ff18::0000:0000 through ff18::ffff:ffff).

	`switch(config)# no cfs ipv6 mcast-address ff15::e244:4754`
	`Distribution over this IP type will be affected`
	`Change multicast address for CFS-IP ?`
	`Are you sure? (y/n) [n] y`
		Reverts to the default IPv6 multicast address for CFS distribution over IPv6. The default IPv6 multicast address for CFS over IP is ff15::efff:4653.

Verifying IP Multicast Address Configuration for CFS over IP

To verify the IP multicast address configuration for CFS over IP, use the **show cfs status** command.

```
switch# show cfs status
Fabric distribution Enabled
IP distribution Enabled mode ipv4
IPv4 multicast address : 10.1.10.100
IPv6 multicast address : ff13::e244:4754
```

Configuring Static IP Peers for CFS over IP

Multicast forwarding is disabled by default in some devices. For example, the IBM Blade chassis has multicast forwarding disabled, especially on external Ethernet ports, and there is no method to enable it. N port virtualization devices use only IP as the transport medium and do not have ISL connectivity or a Fibre Channel domain.

To enable CFS over IP on the switches that do not support multicast forwarding, multicast forwarding has to be enabled on the Ethernet IP switches all along the network that physically connects the switch. In such cases, you can configure static IP peers for CFS distribution over IP.

CFS uses the list of configured IP addresses to communicate with each peer and learn the peer switch WWN. After learning the peer switch WWN, CFS marks the switch as CFS-capable and triggers application-level merging and database distribution.

The following MDS 9000 features require static IP peer configuration for CFS over IP distribution:

- N port virtualization devices have IP as the communication channel because NPV switches do not have FC domain. NPV devices use CFS over IP as the transport medium.
- FlexAttach virtual pWWN distribution on CFS region 201 that links only the NPV-enabled switches.

To configure a static IP peer address for CFS over IP, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.

Step 2	`switch(config)# cfs static-peers`
	`WARNING: This mode will stop dynamic discovery and rely only on these peers.`
	`Do you want to continue? (y/n) [n] y`
	`switch(config-cfs-static)#`
		Enters CFS static peers configuration mode and disables dynamic discovery of peers using multicast forwarding.

	`switch(config)# no cfs static-peers`
	`WARNING: This mode will disable static IP peer configuration and start dynamic discovery of the peers.`
	`Do you want to continue? (y/n) [n] y`
	`switch(config)#`
		Disables CFS static peer discovery and enables dynamic peer discovery using multicast forwarding on all switches.

Step 3	`switch(config-cfs-static)# ip address 1.2.3.4`
	`switch(config-cfs-static)# ip address 1.2.3.5`
	`switch(config-cfs-static)# end`
	`switch#`
		Adds the IP address to the static peers list and marks the switch as CFS-capable. To display the static IP peers list, use the **show cfs static peers** command.

	`switch(config-cfs-static)# no ip address 1.2.3.3`
	`switch(config-cfs-static)# end`
	`switch#`
		Removes the IP address from the static peers list and moves the switch to dynamic peer discovery using multicast forwarding.

Step 4	`switch# show cfs static peers`	Displays the IP address, WWN, and the status of CFS static peer request: • Discovery Inprogress • Local • Reachable • Unreachable • Local IP not present • Rediscovery and distribution disabled

Note IP address and WWN must be configured on the local switch. If CFS does not receive the local switch information, then CFS cannot start any discovery for peer switches.

Verifying Static IP Peer Configuration

To verify the IP peer configuration, use the **show cfs status** command.

```
switch# show cfs status
Distribution: Enabled
Distribution over IP: Enabled - mode IPv4 (static)
IPv4 multicast address : 239:255:70:83
IPv6 multicast address : ff15::efff:4563
```

To display the status of static IP peers disovery, use the **show cfs static peers** command.

```
switch# show cfs static peers
-----------------------------------------------------------
 IP address   WWN name                  Status
-----------------------------------------------------------
 1.2.3.4      00:00:00:00:00:00:00:00   Discovery Inprogress
 1.2.3.5      20:00:00:0d:ec:06:55:b9   Reachable
 1.2.3.6      20:00:00:0d:ec:06:55:c0   Local
```

CFS Regions

This section contains the following topics:

About CFS Regions

A CFS region is a user-defined subset of switches for a given feature or application in its physical distribution scope.When a SAN is spanned across a vast geography, you may need to localize or restrict the distribution of certain profiles among a set of switches based on their physical proximity. Before MDS SAN-OS Release 3.2.(1) the distribution scope of an application within a SAN was spanned across the entire physical fabric without the ability to confine or limit the distribution to a required set of switches in the fabric. CFS regions enables you to overcome this limitation by allowing you to create CFS regions, that is, multiple islands of distribution within the fabric, for a given CFS feature or application. CFS regions are designed to restrict the distribution of a feature's configuration to a specific set or grouping of switches in a fabric.

Note You can only configure a CFS region on physical switches in a SAN. You cannot configure a CFS region in a VSAN.

Example CFS Scenario: Call Home is an application that triggers alerts to Network Administrators when a situation arises or something abnormal occurs. When the fabric covers many geographies and with multiple Network Administrators who are each responsible for a subset of switches in the fabric, the Call Home application sends alerts to all Network Administrators regardless of their location. For the Call Home application to send message alerts selectively to Network Administrators, the physical scope of the application has to be fine tuned or narrowed down, which is achieved by implementing CFS regions.

CFS regions are identified by numbers ranging from 0 through 200. Region 0 is reserved as the default region, and contains every switch in the fabric. You can configure regions from 1 through 200. The default region maintains backward compatibility. If there are switches on the same fabric running releases of SAN-OS before Release 3.2(1), only features in Region 0 are supported when those switches are synchronized. Features from other regions are ignored when those switches are synchronized.

If the feature is moved, that is, assigned to a new region, its scope is restricted to that region; it ignores all other regions for distribution or merging purposes. The assignment of the region to a feature has precedence in distribution over its initial physical scope.

You can configure a CFS region to distribute configurations for multiple features. However, on a given switch, you can configure only one CFS region at a time to distribute the configuration for a given feature. Once you assign a feature to a CFS region, its configuration cannot be distributed within another CFS region.

Managing CFS Regions

This section describes how to manage a CFS region. A set of commands are used to complete the following tasks:

Creating CFS Regions

To create a CFS region, perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# cfs region 4`	Creates a region, for example, number 4.

Assigning Applications to CFS Regions

To assign an applicationon a switch to a region, perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# cfs region 4`	Creates a region, for example, number 4.
Step 3	`switch(config-cfs-region)# ntp` `switch(config-cfs-region)# callhome`	Adds application(s).

Moving an Application to a Different CFS Region

To move an application, for example from Region 1 (originating region) with NTP and Call Home applications to Region 2 (target region), perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# cfs region 2`	Enters the Region 2.
Step 3	`switch(config-cfs-region)# ntp` `switch(config-cfs-region)# callhome`	Indicates application(s) to be moved into Region 2 that originally belong to Region 1. For example, here, the NTP and Call Home applications are moved to Region 2.

Note If you try adding an application to the same region more than once, you see the error message, "Application already present in the same region."

Removing an application from a Region

Removing an application from a region is the same as moving the application back to the default region or to Region 0, that is, bringing the entire fabric into the scope of distribution for the application.

To remove applications from Region 1, perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# cfs region 1`	Enters the Region 1.
Step 3	`switch(config-cfs-region)# no ntp` `switch(config-cfs-region)# no callhome`	Removes application(s) that belong to Region 1, which you want to move.

Deleting CFS Regions

Deleting a region is nullifying the region definition. All the applications bound by the region are released back to the default region by deleting that region.

To delete a region, for example, a region numbered 4, perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.

Step 2	`switch(config)# no cfs region 4`
	`WARNING: All applications in the region wiil be moved to default region.`
	`Are you sure? (y/n) [n]`
		Deletes the Region 4.

Note After Step 2, you see the warning, “All the applications in the region will be moved to the default region.”

Displaying CFS Regions

To display the CFS regions, perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# show cfs region brief`	Displays brief information about the CFS regions.
Step 3	`switch(config)# show cfs region`	Displays detailed information about the CFS regions.

Default Settings

Table 2-1 lists the default settings for CFS configurations.

Table 2-1 Default CFS Parameters

Parameters	Default
CFS distribution on the switch	Enabled.
Database changes	Implicitly enabled with the first configuration change.
Application distribution	Differs based on application.
Commit	Explicit configuration is required.
CFS over IP	Disabled.
IPv4 multicast address	239.255.70.83
IPv6 multicast address	ff15:efff:4653

CHAPTER 3

Configuring System Message Logging

This chapter describes how to configure system message logging on Cisco MDS 9000 Family switches. It includes the following sections:

About System Message Logging

With the system message logging software, you can save messages in a log file or direct the messages to other devices. By default, the switch logs normal but significant system messages to a log file and sends these messages to the system console. This feature provides you with the following capabilities:

- Provides logging information for monitoring and troubleshooting
- Allows you to select the types of captured logging information
- Allows you to select the destination server to forward the captured logging information properly configured system message logging server.

Note When the switch first initializes, the network is not connected until initialization completes. Therefore, messages are not redirected to a system message logging server for a few seconds.

Log messages are not saved across system reboots. However, a maximum of 100 log messages with a severity level of critical and below (levels 0, 1, and 2) are saved in NVRAM.

Table 3-1 describes some samples of the facilities supported by the system message logs.

Table 3-1 Internal Logging Facilities

Facility Keyword	Description	Standard or Cisco MDS Specific
acl	ACL manager	Cisco MDS 9000 Family specific
all	All facilities	Cisco MDS 9000 Family specific
auth	Authorization system	Standard

Table 3-1 Internal Logging Facilities (continued)

Facility Keyword	Description	Standard or Cisco MDS Specific
authpriv	Authorization (private) system	Standard
bootvar	Bootvar	Cisco MDS 9000 Family specific
callhome	Call Home	Cisco MDS 9000 Family specific
cron	Cron or at facility	Standard
daemon	System daemons	Standard
fcc	FCC	Cisco MDS 9000 Family specific
fcdomain	fcdomain	Cisco MDS 9000 Family specific
fcns	Name server	Cisco MDS 9000 Family specific
fcs	FCS	Cisco MDS 9000 Family specific
flogi	FLOGI	Cisco MDS 9000 Family specific
fspf	FSPF	Cisco MDS 9000 Family specific
ftp	File Transfer Protocol	Standard
ipconf	IP configuration	Cisco MDS 9000 Family specific
ipfc	IPFC	Cisco MDS 9000 Family specific
kernel	Kernel	Standard
local0 to local7	Locally defined messages	Standard
lpr	Line printer system	Standard
mail	Mail system	Standard
mcast	Multicast	Cisco MDS 9000 Family specific
module	Switching module	Cisco MDS 9000 Family specific
news	USENET news	Standard
ntp	NTP	Cisco MDS 9000 Family specific
platform	Platform manager	Cisco MDS 9000 Family specific
port	Port	Cisco MDS 9000 Family specific
port-channel	PortChannel	Cisco MDS 9000 Family specific
qos	QoS	Cisco MDS 9000 Family specific
rdl	RDL	Cisco MDS 9000 Family specific
rib	RIB	Cisco MDS 9000 Family specific
rscn	RSCN	Cisco MDS 9000 Family specific
securityd	Security	Cisco MDS 9000 Family specific
syslog	Internal system messages	Standard
sysmgr	System manager	Cisco MDS 9000 Family specific
tlport	TL port	Cisco MDS 9000 Family specific
user	User process	Standard
uucp	UNIX-to-UNIX Copy Program	Standard
vhbad	Virtual host base adapter daemon	Cisco MDS 9000 Family specific

Table 3-1 Internal Logging Facilities (continued)

Facility Keyword	Description	Standard or Cisco MDS Specific
vni	Virtual network interface	Cisco MDS 9000 Family specific
vrrp_cfg	VRRP configuration	Cisco MDS 9000 Family specific
vrrp_eng	VRRP engine	Cisco MDS 9000 Family specific
vsan	VSAN system messages	Cisco MDS 9000 Family specific
vshd	vshd	Cisco MDS 9000 Family specific
wwn	WWN manager	Cisco MDS 9000 Family specific
xbar	Xbar system messages	Cisco MDS 9000 Family specific
zone	Zone server	Cisco MDS 9000 Family specific

Table 3-2 describes the severity levels supported by the system message logs.

Table 3-2 Error Message Severity Levels

Level Keyword	Level	Description	System Message Definition
emergencies	0	System unusable	LOG_EMERG
alerts	1	Immediate action needed	LOG_ALERT
critical	2	Critical conditions	LOG_CRIT
errors	3	Error conditions	LOG_ERR
warnings	4	Warning conditions	LOG_WARNING
notifications	5	Normal but significant condition	LOG_NOTICE
informational	6	Informational messages only	LOG_INFO
debugging	7	Debugging messages	LOG_DEBUG

Note Refer to the *Cisco MDS 9000 Family System Messages Reference* for details on the error log message format.

System Message Logging Configuration

System logging messages are sent to the console based on the default (or configured) logging facility and severity values.

This sections includes the following topics:

- System Message Logging Servers, page 3-6

Message Logging Initiation

You can disable logging to the console or enable logging to a specific Telnet or SSH session.

- When you disable or enable logging to a console session, that state is applied to all future console sessions. If you exit and log in again to a new session, the state is preserved.
- When you enable or disable logging to a Telnet or SSH session, that state is applied only to that session. If you exit and log in again to a new session, the state is not preserved.

To enable or disable the logging state for a Telnet or SSH session, follow these steps: :

	Command	Purpose
Step 1	`switch# terminal monitor`	Enables logging for a Telnet or SSH session. **Note** A console session is enabled by default.
Step 2	`switch# terminal no monitor`	Disables logging for a Telnet or SSH session. **Note** A Telnet or SSH session is disabled by default.

Console Severity Level

When logging is enabled for a console session (default), you can configure the severity levels of messages that appear on the console. The default severity for console logging is 2 (critical).

Tip The current critical (default) logging level is maintained if the console baud speed is 9600 baud (default). All attempts to change the console logging level generates an error message. To increase the logging level (above critical), you must change the console baud speed to 38400 baud.

To configure the severity level for the console session, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# logging console 3`	Configures console logging at level 3 (error). Logging messages with a severity level of 3 or above are displayed on the console.
	`switch(config)# no logging console`	Reverts console logging to the factory set default severity level of 2 (critical). Logging messages with a severity level of 2 or above are displayed on the console.

Monitor Severity Level

When logging is enabled for a monitor session (default), you can configure the severity levels of messages that appear on the monitor. The default severity for monitor logging is 5 (notifications).

To configure the severity level for a monitor session, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# logging monitor 3`	Configures monitor logging at level 3 (error). Logging messages with a severity level of 3 or above are displayed on the monitor.
	`switch(config)# no logging monitor`	Reverts monitor logging to the factory set default severity level of 5 (notifications). Logging messages with a severity level of 5 or above are displayed on the console.

Module Logging

By default, logging is enabled at level 7 for all modules. You can enable or disable logging for each module at a specified level.

To enable or disable the logging for modules and configure the severity level, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# logging module 1`	Configures module logging at level 1 (alerts) for all modules.
	`switch(config)# logging module`	Configures module logging for all modules in the switch at the default level 5 (notifications).
	`switch(config)# no logging module`	Disables module logging.

Facility Severity Levels

To configure the severity level for a logging facility (see Table 3-1), follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# logging level kernel 4`	Configures Telnet or SSH logging for the kernel facility at level 4 (warning). As a result, logging messages with a severity level of 4 or above are displayed.
	`switch(config)# no logging level kernel 4`	Reverts to the default severity level 6 (informational) for the Telnet or SSH logging for the kernel facility. **Note** Use the **show logging info** command to display the default logging levels for the facilities listed in Table 3-1.

Log Files

By default, the switch logs normal but significant system messages to a log file and sends these messages to the system console. Log messages are not saved across system reboots.The logging messages that are generated may be saved to a log file. You can configure the name of this file and restrict its size as required. The default log file name is messages. The file name can have up to 80 characters and the file size ranges from 4096 bytes to 4194304 bytes.

To send log messages to a file, follow these steps: :

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# logging logfile messages 3`	Configures logging of information for errors or events above with a severity level 3 or above to the default log file named messages.
	`switch(config)# logging logfile ManagerLog 3`	Configures logging of information for errors or events with a severity level 3 or above to a file named ManagerLog using the default size of 10,485,760 bytes.
	`switch(config)# logging logfile ManagerLog 3 size 3000000`	Configures logging information for errors or events with a severity level 3 or above to a file named ManagerLog. By configuring a size, you are restricting the file size to 3,000,000 bytes.
	`switch(config)# no logging logfile`	Disables logging messages to the logfile.

Note You can rename the log file using the **logging logfile** command.

The configured log file is saved in the /var/log/external directory. The location of the log file cannot be changed. You can use the **show logging logfile** and **clear logging logfile** commands to view and delete the contents of this file. You can use the **dir log:** command to view logging file statistics. You can use the **delete log:** command to remove the log file.

You can copy the logfile to a different location using the **copy log:** command using additional copy syntax.

System Message Logging Servers

You can configure a maximum of three system message logging servers.

To send log messages to a UNIX system message logging server, you must configure the system message logging daemon on a UNIX server. Log in as root, and follow these steps:

Step 1 Add the following line to the /etc/syslog.conf file.

local1.debug **/var/log/***myfile***.log**

Note Be sure to add five tab characters between **local1.debug** and **/var/log/***myfile***.log**. Refer to entries in the /etc/syslog.conf file for further examples.

The switch sends messages according to the specified facility types and severity levels. The **local1** keyword specifies the UNIX logging facility used. The messages from the switch are generated by user processes. The **debug** keyword specifies the severity level of the condition being logged. You can set UNIX systems to receive all messages from the switch.

Step 2 Create the log file by entering these commands at the UNIX shell prompt:

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

Step 3 Make sure the system message logging daemon reads the new changes by entering this command:

```
$ kill -HUP ~cat /etc/syslog.pid~
```

To configure system message logging server IPv4 addresses, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch#`	Enters configuration mode.
Step 2	`switch(config)# logging server 172.22.00.00`	Configures the switch to forward log messages according to the specified facility types and severity levels to remote multiple servers specified by its hostname or IPv4 address (172.22.00.00).
	`switch(config)# logging server 172.22.00.00 facility local1`	Configures the switch to forward log messages according to the specified facility (local1) for the server IPv4 address (172.22.00.00). The default outgoing facility is local7.
	`switch(config)# no logging server 172.11.00.00`	Removes the specified server (172.11.00.00) and reverts to factory default.

To configure system message logging server IPv6 addresses, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch#`	Enters configuration mode.
Step 2	`switch(config)# logging server 2001::0db8:800:200c:417a`	Configures the switch to forward log messages according to the specified facility types and severity levels to a remote server specified by its IPv6 address.
	`switch(config)# logging server 2001::0db8:800:200c:417a facility local1`	Configures the switch to forward log messages according to the specified facility (local1) for the server IPv6 address. The default outgoing facility is local7.
	`switch(config)# no logging server 2001::0db8:800:200c:417a`	Removes the specified server and reverts to factory default.

System Message Logging

The system message logging software saves the messages in a log file or directs the messages to other devices. This feature has the following capabilities:

- Provides logging information for monitoring and troubleshooting.
- Allows the user to select the types of captured logging information.
- Allows the user to select the destination server to forward the captured logging information.

By default, the switch logs normal but significant system messages to a log file and sends these messages to the system console. You can specify which system messages should be saved based on the type of facility and the severity level. Messages are time-stamped to enhance real-time debugging and management.

You can access the logged system messages using the CLI or by saving them to a correctly configured system message logging server. The switch software saves system messages in a file that can save up to 1200 entries. You can monitor system messages remotely by accessing the switch through Telnet, SSH, ,the console port, or by viewing the logs on a system message logging server.

Outgoing System Message Logging Server Facilities

All system messages have a logging facility and a level. The logging facility can be thought of as *where* and the level can be thought of as *what*.

The single system message logging daemon (syslogd) sends the information based on the configured **facility** option. If no facility is specified, local7 is the default outgoing facility.

The internal facilities are listed in Table 3-1 and the outgoing logging facilities are listed in Table 3-3.

Table 3-3 Outgoing Logging Facilities

Facility Keyword	Description	Standard or Cisco MDS Specific
auth	Authorization system	Standard
authpriv	Authorization (private) system	Standard
cron	Cron or at facility	Standard
daemon	System daemons	Standard
ftp	File Transfer Protocol	Standard
kernel	Kernel	Standard
local0 to local7	Locally defined messages	Standard (local7 is the default)
lpr	Line printer system	Standard
mail	Mail system	Standard
news	USENET news	Standard
syslog	Internal system messages	Standard
user	User process	Standard
uucp	UNIX-to-UNIX Copy Program	Standard

System Message Logging Configuration Distribution

You can enable fabric distribution for all Cisco MDS switches in the fabric. When you perform system message logging configurations, and distribution is enabled, that configuration is distributed to all the switches in the fabric.

You automatically acquire a fabric-wide lock when you issue the first configuration command after you enabled distribution in a switch. The system message logging server uses the effective and pending database model to store or commit the commands based on your configuration. When you commit the configuration changes, the effective database is overwritten by the configuration changes in the pending database and all the switches in the fabric receive the same configuration. After making the configuration changes, you can choose to discard the changes by aborting the changes instead of committing them. In either case, the lock is released. See Chapter 2, “Using the CFS Infrastructure” for more information on the CFS application.

To enable fabric distribution for system message logging server configurations, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# logging distribute`	Enables the system message logging server configuration to be distributed to all switches in the fabric, acquires a lock, and stores all future configuration changes in the pending database.
	`switch(config)# no logging distribute`	Disables (default) system message logging server configuration distribution to all switches in the fabric.

To commit the system message logging server configuration changes, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# logging commit`	Distributes the configuration changes to all switches in the fabric, releases the lock, and overwrites the effective database with the changes made to the pending database.

To discard the system message logging server configuration changes, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# logging abort`	Discards the system message logging server configuration changes in the pending database and releases the fabric lock.

Fabric Lock Override

If you have performed a system message logging task and have forgotten to release the lock by either committing or discarding the changes, an administrator can release the lock from any switch in the fabric. If the administrator performs this task, your changes to the pending database are discarded and the fabric lock is released.

Tip The changes are only available in the volatile directory and are subject to being discarded if the switch is restarted.

To use administrative privileges and release a locked system message logging session, use the **clear logging session** command.

```
switch# clear logging session
```

Database Merge Guidelines

See the "CFS Merge Support" section on page 2-9 for detailed concepts.

When merging two system message logging databases, follow these guidelines:

- Be aware that the merged database is a union of the existing and received database for each switch in the fabric.
- Verify that the merged database will only have a maximum of three system message logging servers.

Caution If the merged database contains more that three servers, the merge will fail.

Displaying System Message Logging Information

Use the **show logging** command to display the current system message logging configuration. See Examples 3-1 to 3-10.

Note When using the **show logging** command, output is displayed only when the configured logging levels for the switch are different from the default levels.

Example 3-1 Displays Current System Message Logging

```
switch# show logging
Logging console:                 enabled (Severity: critical)
Logging monitor:                 enabled (Severity: debugging)
Logging linecard:                enabled (Severity: debugging)
Logging server:                  enabled
{172.20.102.34}
        server severity:         debugging
        server facility:         local7
{10.77.202.88}
        server severity:         debugging
        server facility:         local7
{10.77.202.149}
        server severity:         debugging
        server facility:         local7
Logging logfile:                 enabled
        Name - messages: Severity - debugging Size - 4194304
Facility        Default Severity        Current Session Severity
--------        ----------------        ------------------------
kern                    6                       6
user                    3                       3
mail                    3                       3
```

```
daemon                  7                       7
auth                    0                       7
syslog                  3                       3
lpr                     3                       3
news                    3                       3
uucp                    3                       3
cron                    3                       3
authpriv                3                       7
ftp                     3                       3
local0                  3                       3
local1                  3                       3
local2                  3                       3
local3                  3                       3
local4                  3                       3
local5                  3                       3
local6                  3                       3
local7                  3                       3
vsan                    2                       2
fspf                    3                       3
fcdomain                2                       2
module                  5                       5
sysmgr                  3                       3
zone                    2                       2
vni                     2                       2
ipconf                  2                       2
ipfc                    2                       2
xbar                    3                       3
fcns                    2                       2
fcs                     2                       2
acl                     2                       2
tlport                  2                       2
port                    5                       5
flogi                   2                       2
port_channel            5                       5
wwn                     3                       3
fcc                     2                       2
qos                     3                       3
vrrp_cfg                2                       2
ntp                     2                       2
platform                5                       5
vrrp_eng                2                       2
callhome                2                       2
mcast                   2                       2
rdl                     2                       2
rscn                    2                       2
bootvar                 5                       2
securityd               2                       2
vhbad                   2                       2
rib                     2                       2
vshd                    5                       5
0(emergencies)          1(alerts)       2(critical)
3(errors)               4(warnings)     5(notifications)
6(information)          7(debugging)

Feb 14 09:50:57 excal-113 %TTYD-6-TTYD_MISC: TTYD TTYD started
Feb 14 09:50:58 excal-113 %DAEMON-6-SYSTEM_MSG: precision = 8 usec
...
```

Use the **show logging nvram** command to view the log messages saved in NVRAM. Only log messages with a severity level of critical and below (levels 0, 1, and 2) are saved in NVRAM.

Example 3-2 Displays NVRM Log Contents

```
switch# show logging nvram
Jul 16 20:36:46 172.22.91.204 %KERN-2-SYSTEM_MSG: unable to alloc and fill in a
new mtsbuf (pid=2209, ret_val = -105)
Jul 16 20:36:46 172.22.91.204 %KERN-2-SYSTEM_MSG: unable to alloc and fill in a
new mtsbuf (pid=2199, ret_val = -105)
Jul 16 20:36:46 172.22.91.204 %KERN-2-SYSTEM_MSG: unable to alloc and fill in a
new mtsbuf (pid=2213, ret_val = -105)
Jul 16 20:36:46 172.22.91.204 %KERN-2-SYSTEM_MSG: unable to alloc and fill in a
new mtsbuf (pid=2213, ret_val = -105)
...
```

Example 3-3 Displays the Log File

```
switch# show logging logfile
Jul 16 21:06:50 %DAEMON-3-SYSTEM_MSG: Un-parsable frequency in /mnt/pss/ntp.drift
Jul 16 21:06:56 %DAEMON-3-SYSTEM_MSG: snmpd:snmp_open_debug_cfg: no snmp_saved_dbg_uri ;
Jul 16 21:06:58 172.22.91.204 %PORT-5-IF_UP: Interface mgmt0 is up
Jul 16 21:06:58 172.22.91.204 %MODULE-5-ACTIVE_SUP_OK: Supervisor 5 is active
...
```

Example 3-4 Displays Console Logging Status

```
switch# show logging console
Logging console:                enabled (Severity: notifications)
```

Example 3-5 Displays Logging Facility

```
switch# show logging level
Facility        Default Severity        Current Session Severity
--------        ----------------        ------------------------
kern                    6                       6
user                    3                       3
mail                    3                       3
daemon                  7                       7
auth                    0                       7
syslog                  3                       3
lpr                     3                       3
news                    3                       3
uucp                    3                       3
cron                    3                       3
authpriv                3                       7
ftp                     3                       3
local0                  3                       3
local1                  3                       3
local2                  3                       3
local3                  3                       3
local4                  3                       3
local5                  3                       3
local6                  3                       3
local7                  3                       3
vsan                    2                       2
fspf                    3                       3
fcdomain                2                       2
module                  5                       5
sysmgr                  3                       3
zone                    2                       2
vni                     2                       2
ipconf                  2                       2
ipfc                    2                       2
```

```
xbar                    3                       3
fcns                    2                       2
fcs                     2                       2
acl                     2                       2
tlport                  2                       2
port                    5                       5
flogi                   2                       2
port_channel            5                       5
wwn                     3                       3
fcc                     2                       2
qos                     3                       3
vrrp_cfg                2                       2
ntp                     2                       2
platform                5                       5
vrrp_eng                2                       2
callhome                2                       2
mcast                   2                       2
rdl                     2                       2
rscn                    2                       2
bootvar                 5                       2
securityd               2                       2
vhbad                   2                       2
rib                     2                       2
vshd                    5                       5
0(emergencies)          1(alerts)       2(critical)
3(errors)               4(warnings)     5(notifications)
6(information)          7(debugging)
```

Example 3-6 Displays Logging Information

```
switch# show logging info
Logging console:                enabled (Severity: critical)
Logging monitor:                enabled (Severity: debugging)
Logging linecard:               enabled (Severity: debugging)
Logging server:                 enabled
{172.20.102.34}
        server severity:        debugging
        server facility:        local7
{10.77.202.88}
        server severity:        debugging
        server facility:        local7
{10.77.202.149}
        server severity:        debugging
        server facility:        local7
Logging logfile:                enabled
        Name - messages: Severity - debugging Size - 4194304
Facility        Default Severity        Current Session Severity
--------        ----------------        ------------------------
kern                    6                       6
user                    3                       3
mail                    3                       3
daemon                  7                       7
auth                    0                       7
syslog                  3                       3
lpr                     3                       3
news                    3                       3
uucp                    3                       3
cron                    3                       3
authpriv                3                       7
ftp                     3                       3
local0                  3                       3
local1                  3                       3
local2                  3                       3
```

```
local3                  3                       3
local4                  3                       3
local5                  3                       3
local6                  3                       3
local7                  3                       3
vsan                    2                       2
fspf                    3                       3
fcdomain                2                       2
module                  5                       5
sysmgr                  3                       3
zone                    2                       2
vni                     2                       2
ipconf                  2                       2
ipfc                    2                       2
xbar                    3                       3
fcns                    2                       2
fcs                     2                       2
acl                     2                       2
tlport                  2                       2
port                    5                       5
flogi                   2                       2
port_channel            5                       5
wwn                     3                       3
fcc                     2                       2
qos                     3                       3
vrrp_cfg                2                       2
ntp                     2                       2
platform                5                       5
vrrp_eng                2                       2
callhome                2                       2
mcast                   2                       2
rdl                     2                       2
rscn                    2                       2
bootvar                 5                       2
securityd               2                       2
vhbad                   2                       2
rib                     2                       2
vshd                    5                       5
0(emergencies)          1(alerts)       2(critical)
3(errors)               4(warnings)     5(notifications)
6(information)          7(debugging)
```

Example 3-7 Displays Last Few Lines of a Log File

```
switch# show logging last 2
Nov 8 16:48:04 excal-113 %LOG_VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from pts/1
(171.71.58.56)
Nov 8 17:44:09 excal-113 %LOG_VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from pts/0
(171.71.58.72)
```

Example 3-8 Displays Switching Module Logging Status

```
switch# show logging module
Logging linecard:                enabled (Severity: debugging)
```

Example 3-9 Displays Monitor Logging Status

```
switch# show logging monitor
Logging monitor:                 enabled (Severity: information)
```

Example 3-10 Displays Server Information

```
switch# show logging server
Logging server:                 enabled
{172.22.95.167}
        server severity:        debugging
        server facility:        local7
{172.22.92.58}
        server severity:        debugging
        server facility:        local7
```

Default Settings

Table 3-4 lists the default settings for system message logging.

Table 3-4 Default System Message Log Settings

Parameters	Default
System message logging to the console	Enabled for messages at the critical severity level.
System message logging to Telnet sessions	Disabled.
Logging file size	4194304.
Log file name	Message (change to a name with up to 200 characters).
Logging server	Disabled.
Syslog server IP address	Not configured.
Number of servers	Three servers.
Server facility	Local 7.

CHAPTER 4

Configuring Call Home

Call Home provides e-mail-based notification of critical system events. A versatile range of message formats are available for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications. Common uses of this feature may include direct paging of a network support engineer, e-mail notification to a Network Operations Center, and utilization of Cisco Smart Call Home services for direct case generation with the Technical Assistance Center.

Note Cisco Autonotify is upgraded to a new capability called Smart Call Home. Smart Call Home has significant functionality improvement over Autonotify and is available across the Cisco product range. For detailed information on Smart Call Home, see the Smart Call Home page at this location: http://www.cisco.com/go/smartcall/

The Call Home feature provides message throttling capabilities. Periodic inventory messages, port syslog messages, and RMON alert messages are added to the list of deliverable Call Home messages. If required you can also use the Cisco Fabric Services application to distribute the Call Home configuration to all other switches in the fabric.

This chapter includes the following sections:

About Call Home

The Call Home service provides e-mail-based notification of critical system events. A versatile range of message formats are available for optimal compatibility with pager services, standard e-mail, or XML-based automated parsing applications.

Common features may include the following:

- Paging the network support engineer
- E-mailing the Network Operations Center
- Raising a direct case with the Technical Assistance Center

The Call Home functionality is available directly through the Cisco MDS 9000 Family switches and the Cisco Nexus 5000 Series switches. It provides multiple Call Home messages, each with separate potential destinations. You can define your own destination profiles in addition to predefined profiles; you can configure up to 50 e-mail addresses for each destination profile. Flexible message delivery and format options make it easy to integrate specific support requirements.

The Call Home feature offers the following advantages:

- Fixed set of predefined alerts for trigger events on the switch.
- Automatic execution and attachment of relevant command output.

Call Home Features

The Call Home functionality is available directly through the Cisco MDS 9000 Family switches and the Cisco Nexus 5000 Series switches. It provides multiple Call Home profiles (also referred to as *Call Home destination profiles*), each with separate potential destinations. You can define your own destination profiles in addition to predefined profiles.

The Call Home function can even leverage support from Cisco Systems or another support partner. Flexible message delivery and format options make it easy to integrate specific support requirements.

The Call Home feature offers the following advantages:

- Fixed set of predefined alerts and trigger events on the switch.
- Automatic execution and attachment of relevant command output.

- Multiple message format options:
  - Short Text—Suitable for pagers or printed reports.
  - Plain Text—Full formatted message information suitable for human reading.
  - XML—Matching readable format using Extensible Markup Language (XML) and document type definitions (DTDs) named Messaging Markup Language (MML). The MML DTD is published on the Cisco.com website at http://www.cisco.com/. The XML format enables communication with the Cisco Systems Technical Assistance Center.
- Multiple concurrent message destinations. You can configure up to 50 e-mail destination addresses for each destination profile.
- Multiple message categories including system, environment, switching module hardware, supervisor module, hardware, inventory, syslog, RMON, and test.

About Smart Call Home

Smart Call Home is a component of Cisco SMARTnet Service that offers proactive diagnostics, real-time alerts, and personalized web-based reports on select Cisco devices.

Smart Call Home provides fast resolution of system problems by analyzing Call Home messages sent from your devices and providing a direct notification path to Cisco customer support.

Smart Call Home offers the following features:

- Continuous device health monitoring and real-time diagnostics alerts.
- Analysis of Call Home messages from your device and where appropriate, automatic service request generation, routed to the appropriate TAC team, including detailed diagnostic information to speed problem resolution.
- Secure message transport through a downloadable Transport Gateway (TG) aggregation point. You can use a TG aggregation point in cases requiring support for multiple devices or in cases where security requirements mandate that your devices not be connected directly to the Internet.
- Web-based access to Call Home messages and recommendations, inventory and configuration information for all Call Home devices. Provides access to associated Field Notices, Security Advisories and End-of-Life Information.

Table 4-1 lists the benefits of Smart Call Home.

Table 4-1 Benefits of Smart Call Home Compared to Autonotify

Feature	Smart Call Home	Autonotify
Low touch registration	The registration process is considerably streamlined. Customers no longer need to know their device serial number or contract information. They can register devices without manual intervention from Cisco by sending a message from those devices. The procedures are outlined at www.cisco.com/go/smartcall	Requires the customer to request Cisco to add each specific serial number to the database.
Recommendations	Smart Call Home provides recommendations for known issues including those for which SRs are raised and for which SRs are not appropriate but for which customers might want to still take action on.	Autonotify raises SRs for a set of failure scenarios but no recommendations are provided for these.
Device report	Device report includes full inventory and configuration details. Once available, the information in these reports will be mapped to field notices, PSIRTs, EoX notices, configuration best practices and bugs.	No.
History report	The history report is available to look up any message and its contents, including **show** commands, message processing, analysis results, recommendations and service request numbers for all messages sent over the past three months.	A basic version is available that does not include contents of message.

Table 4-1 Benefits of Smart Call Home Compared to Autonotify (continued)

Feature	Smart Call Home	Autonotify
Network summary report	A report that provides a summary of the make-up of devices and modules in the customer network (for those devices registered with Smart Call home)	No.
Cisco device support	Device Support will be extended across the Cisco product range. See the supported products table at www.cisco.com/go/smartcall	Deprecated in favor of Smart Call Home in October 2008.

Obtaining Smart Call Home

If you have a service contract directly with Cisco Systems, you can receive automatic case generation from the Technical Assistance Center by registering with the Smart Call Home service.

You need the following items to register:

- The SMARTnet contract number for your switch.
- Your e-mail address
- Your Cisco.com ID

For detailed information on Smart Call Home, including quick start configuration and registration steps, see the Smart Call Home page at this location:

http://www.cisco.com/go/smartcall/

Configuring Call Home

How you configure the Call Home process depends on how you intend to use the feature. Some points to consider include:

- An e-mail server and at least one destination profile (predefined or user-defined) must be configured. The destination profile(s) used depends on whether the receiving entity is a pager, e-mail, or automated service such as Cisco Smart Call Home.
- Switches can forward events (SNMP traps/informs) up to 10 destinations.
- The contact name (SNMP server contact), phone, and street address information must be configured before Call Home is enabled. This configuration is required to determine the origin of messages received.
- The Cisco MDS 9000 Family switch and the Cisco Nexus 5000 Series switch must have IP connectivity to an e-mail server.
- If Cisco Smart Call Home is used, an active service contract must cover the device being configured.

To configure Call Home, follow these steps:

Step 1 Assign contact information.

Step 2 Configure destination profiles.

Step 3 Associate one or more alert groups to each profile as required by your network. Customize the alert groups, if desired.

Step 4 Configure e-mail options.

Step 5 Enable or disable Call Home.

Step 6 Test Call Home messages.

Configuring Contact Information

Each switch must include e-mail, phone, and street address information. You can optionally include the contract ID, customer ID, site ID, and switch priority information.

Note Switch priority is specific to each switch in the fabric. This priority is used by the operatFions personnel or TAC support personnel to decide which Call Home message they should respond to first. You can prioritize Call Home alerts of the same severity from each switch.

To assign the contact information, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# snmp-server contact personname@companyname.com`	Configures the SNMP contact name.
Step 3	`switch(config)# callhome` `switch(config-callhome)#`	Enters the Call Home configuration submode.
Step 4	`switch(config-callhome)# e-mail-contact username@company.com`	Assigns the customer's e-mail address. Up to 128 alphanumeric characters are accepted in e-mail address format. **Note** You can use any valid e-mail address. You cannot use spaces.
Step 5	`switch(config-callhome)# phone-contact +1-800-123-4567`	Assigns the customer's phone number. Up to 20 alphanumeric characters are accepted in international format. **Note** You cannot use spaces. Be sure to use the + prefix before the number.
Step 6	`switch(config-callhome)# streetaddress 1234 Picaboo Street, Any city, Any state, 12345`	Assigns the customer's street address where the equipment is located. Up to 256 alphanumeric characters are accepted in free format.

	Command	Purpose
Step 7	`switch(config-callhome)# switch-priority 0`	Assigns the switch priority, with 0 being the highest priority and 7 the lowest. **Tip** Use this field to create a hierarchical management structure.
Step 8	`switch(config-callhome)# customer-id Customer1234`	Optional. Identifies the customer ID. Up to 256 alphanumeric characters are accepted in free format.
Step 9	`switch(config-callhome)# site-id Site1ManhattanNY`	Optional. Identifies the customer site ID. Up to 256 alphanumeric characters are accepted in free format.
Step 10	`switch(config-callhome)# contract-id Company1234`	Assigns the customer ID for the switch. Up to 64 alphanumeric characters are accepted in free format.

Destination Profiles

A destination profile contains the required delivery information for an alert notification. Destination profiles are typically configured by the network administrator. At least one destination profile is required. You can configure multiple destination profiles of one or more types. You can use one of the predefined destination profiles or define a desired profile. If you define a new profile, you must assign a profile name.

Using alert groups you can select the set of Call Home alerts to be received by a destination profile (predefined or user defined). Alert groups are predefined subsets of Call Home alerts supported in all switches in the Cisco MDS 9000 Family and the Cisco Nexus 5000 Series. Different types of Call Home alerts are grouped into different alert groups depending on their type. You can associate one or more alert groups to each profile as required by your network.

Note If you use the Cisco Smart Call Home service, the XML destination profile is required (see http://www.cisco.com/en/US/partner/products/hw/ps4159/ps4358/products_configuration_example09186a0080108e72.shtml).

You can configure the following attributes for a destination profile:

- Profile name—A string that uniquely identifies each user-defined destination profile and is limited to 32 alphanumeric characters. The format options for a user-defined destination profile are full-txt, short-txt, or XML (default).
- Destination address—The actual address, pertinent to the transport mechanism, to which the alert should be sent.
- Message formatting—The message format used for sending the alert (full text, short text, or XML).

Configuring Destination Profiles

To configure predefined destination profile messaging options, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters the Call Home configuration submode.
Step 3	`switch(config-callhome)# destination-profile full-txt-destination e-mail-addr person@place.com`	Configures an e-mail address for the predefined full-txt-destination profile. The e-mail addresses in this destination profile receives messages in full-txt format. The full-text format provides the complete, detailed explanation of the failure. **Tip** Use a standard e-mail address that does not have any text size restrictions.
	`switch(config-callhome)# destination-profile full-txt-destination message-size 1000000`	Configures a maximum destination message size for the predefined full-txt-destination profile. The valid range is 0 to 1,000,000 bytes and the default is 500,000. A value of 0 implies that a message of any size can be sent.
Step 4	`switch(config-callhome)# destination-profile short-txt-destination e-mail-addr person@place.com`	Configures an e-mail address for the predefined short-txt-destination profile. The e-mail addresses in this destination profile receive messages in short-txt format. This format provides the basic explanation of the failure in the Call Home message. **Tip** Use a pager-related e-mail address for this option.
	`switch(config-callhome)# destination-profile short-txt-destination message-size 100000`	Configures maximum destination message size for the predefined short-txt-destination profile. The valid range is 0 to 1,000,000 bytes and the default is 4000. A value of 0 implies that a message of any size can be sent.
Step 5	`switch(config-callhome)# destination-profile XML-destination e-mail-addr findout@.cisco.com`	Configures an e-mail address for the predefined XML-destination profile. The e-mail addresses in this destination-profile receives messages in XML format. This format provides information that is compatible with Cisco Systems TAC support. **Tip** Do not add a pager-related e-mail address to this destination profile because of the large message size.
	`switch(config-callhome)# destination-profile XML-destination message-size 100000`	Configures maximum destination message size for the predefined destination profile XML-destination. The valid range is 0 to 1,000,000 bytes and the default is 500,000. A value of 0 implies that a message of any size can be sent.

Note Steps 3, 4, and 5 in this procedure can be skipped or configured in any order.

To configure a new destination-profile (and related parameters), follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters the Call Home configuration submode.
Step 3	`switch(config-callhome)# destination-profile test`	Configures a new destination profile called test.
Step 4	`switch(config-callhome)# destination-profile test e-mail-addr person@place.com`	Configures the e-mail address for the user-defined destination profile (test) sent in default XML format.
Step 5	`switch(config-callhome)# destination-profile test message-size 1000000`	Configures a maximum message size for the destination e-mail addresses in the user-defined destination profile (test) sent in default XML format. The valid range is 0 to 1,000,000 bytes and the default is 500,000. A value of 0 implies that a message of any size can be sent.
Step 6	`switch(config-callhome)# destination-profile test format full-txt`	Configures message-format for the user-defined destination profile (test) to be full text format.
	`switch(config-callhome)# destination-profile test format short-txt`	Configures message-format for the user-defined destination profile (test) to be short text format.

Note Steps 4, 5, and 6 in this procedure can be skipped or configured in any order.

Call Home Alert Groups

An alert group is a predefined subset of Call Home alerts supported in all switches in the Cisco MDS 9000 Family and Cisco Nexus 5000 Series. Alert groups allow you to select the set of Call Home alerts to be received by a destination profile (predefined or user-defined). A Call Home alert is sent to e-mail destinations in a destination profile only if that Call Home alert belongs to one of the alert groups associated with that destination profile.

Using the predefined Call Home alert groups you can generate notification messages when certain events occur on the switch. You can customize predefined alert groups to execute additional **show** commands when specific events occur and to notify you of output other than from the predefined **show** commands.

Different types of Call Home alerts are grouped into different alert groups depending on their type. You can associate one or more alert groups to each profile as required by your network.

The alert group feature allows you to select the set of Call Home alerts to be received by a destination profile (either predefined or user-defined). You can associate multiple alert groups with a destination profile.

Note A Call Home alert is sent to e-mail destinations in a destination profile only if that Call Home alert belongs to one of the alert groups associated with that destination profile.

Associating an Alert Group

To associate an alert group with a destination profile, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# destination-profile test1 alert-group test`	Optional. Configures user-defined destination profile (test1) to receive all user-generated Call Home test notifications.
	`switch(config-callhome)# destination-profile short-txt-destination alert-group test`	Optional. Configures predefined short-text destination profile to receive all user-generated Call Home test notifications.
Step 4	`switch(config-callhome)# destination-profile test1 alert-group all`	Optional. Configures user-defined destination profile (test1) to receive Call Home notifications for all events
	`switch(config-callhome)# destination-profile short-txt-destination alert-group all`	Optional. Configures predefined short-text destination message profile to receive Call Home notifications for all (default) events
Step 5	`switch(config-callhome)# destination-profile test1 alert-group Cisco-TAC`	Optional. Configures user-defined destination message profile (test1) to receive Call Home notifications for events that are meant only for Cisco TAC or the Auto-notify service.
	`switch(config-callhome)# destination-profile xml-destination alert-group Cisco-TAC`	Optional. Configures predefined XML destination message profile to receive Call Home notifications for events that are meant only for Cisco TAC or the auto-notify service.
Step 6	`switch(config-callhome)# destination-profile test1 alert-group environmental`	Optional. Configures user-defined destination message profile (test1) to receive Call Home notifications for power, fan, and temperature-related events.
	`switch(config-callhome)# destination-profile short-txt-destination alert-group environmental`	Optional. Configures predefined short-text destination message profile to receive Call Home notifications for power, fan, and temperature-related events.
Step 7	`switch(config-callhome)# destination-profile test1 alert-group inventory`	Optional. Configures user-defined destination message profile (test1) to receive Call Home notifications for inventory status events.
	`switch(config-callhome)# destination-profile short-txt-destination alert-group inventory`	Optional. Configures predefined short-text destination message profile to receive Call Home notifications for inventory status events.
Step 8	`switch(config-callhome)# destination-profile test1 alert-group linecard-hardware`	Optional. Configures user-defined destination message profile (test1) to receive Call Home notifications for module-related events.
	`switch(config-callhome)# destination-profile short-txt-destination alert-group linecard-hardware`	Optional. Configures predefined short-text destination message profile to receive Call Home notifications for module-related events.

	Command	Purpose
Step 9	`switch(config-callhome)# destination-profile test1 alert-group supervisor-hardware`	Optional. Configures user-defined destination message profile (test1) to receive Call Home notifications for supervisor-related events.
	`switch(config-callhome)# destination-profile short-txt-destination alert-group supervisor-hardware`	Optional. Configures predefined short-text destination message profile to receive Call Home notifications for supervisor-related events.
Step 10	`switch(config-callhome)# destination-profile test1 alert-group system`	Optional. Configures user-defined destination message profile (test1) to receive Call Home notifications for software-related events.
	`switch(config-callhome)# destination-profile short-txt-destination alert-group system`	Optional. Configures predefined short-text destination message profile to receive Call Home notifications for software-related events.

Customized Alert Group Messages

An alert group is a predefined subset of Call Home alerts supported in all switches in the Cisco MDS 9000 Family and Cisco Nexus 5000 Series. Alert groups allow you to select the set of Call Home alerts to be received by a destination profile (predefined or user-defined). The predefined Call Home alert groups generate notification messages when certain events occur on the switch. You can customize predefined alert groups to execute additional show commands when specific events occur. The output from these additional **show** commands is included in the notification message along with the output of the predefined **show** commands.

Note You can assign a maximum of five user-defined **show** commands to an alert group. Only **show** commands can be assigned to an alert group.

Note Customized **show** commands are only supported for full text and XML alert groups. Short text alert groups (short-txt-destination) do not support customized **show** commands because they only allow 128 bytes of text.

To assign **show** commands to be executed when an alert is sent, you must associate the commands with the alert group. When an alert is sent, Call Home associates the alert group with an alert type and attaches the output of the **show** commands to the alert message.

Note Make sure the destination profiles for a non-Cisco-TAC alert group, with a predefined **show** command, and the Cisco-TAC alert group are not the same.

Customizing Alert Group Messages

To customize Call Home alert group messages, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# alert-group license user-def-cmd show license usage`	Configures a user-defined **show** command for an alert group license. **Note** Only valid **show** commands are accepted.
	`switch(config-callhome)# no alert-group license user-def-cmd show license usage`	Removes the user-defined **show** command from the alert group.

Verifying Alert Group Customization

To verify the alert group customization, use the **show callhome user-def-cmds** command.

```
switch# show callhome user-def-cmds
User configured commands for alert groups :
alert-group test user-def-cmd "show version"
```

Call Home Message Level Feature

The Call Home message level feature allows you to filter messages based on their level of urgency. Each destination profile (predefined and user-defined) is associated with a Call Home message level threshold. Any message with a value lower than the urgency threshold is not sent. The urgency level ranges from 0 (lowest level of urgency) to 9 (highest level of urgency), and the default is 0 (all messages are sent).

Note Call Home severity levels are not the same as system message logging severity levels.

Setting the Call Home Message Levels

To set the message level for each destination profile for Call Home, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# destination-profile test message-level 5`	Optional. Configures the message level urgency as 5 and above for the user-defined profile (test1).
	`switch(config-callhome)# no destination-profile oldtest message-level 7`	Removes a previously configured urgency level and reverts it to the default of 0 (all messages are sent).

Syslog-Based Alerts

You can configure the switch to send certain syslog messages as Call Home messages. The messages are sent based on the mapping between the destination profile and the alert group mapping, and on the severity level of the generated syslog message.

To receive a syslog-based Call Home alert, you must associate a destination profile with the syslog alert groups (currently there is only one syslog alert group—syslog-group-port) and configure the appropriate message level.

The syslog-group-port alert group selects syslog messages for the port facility. The Call Home application maps the syslog severity level to the corresponding Call Home severity level (see the "Call Home Message Levels" section on page 4-34). For example, if you select level 5 for the Call Home message level, syslog messages at levels 0, 1, and 2 are included in the Call Home log.

Whenever a syslog message is generated, the Call Home application sends a Call Home message depending on the mapping between the destination profile and the alert group mapping and based on the severity level of the generated syslog message. To receive a syslog-based Call Home alert, you must associate a destination profile with the syslog alert groups (currently there is only one syslog alert group—syslog-group-port) and configure the appropriate message level (see the "Call Home Message Level Feature" section on page 4-12).

Note Call Home does not change the syslog message level in the message text. The syslog message texts in the Call Home log appear as they are described in the *Cisco MDS 9000 Family System Messages Reference*.

Configuring the Syslog-Based Alerts

To configure the syslog-group-port alert group, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# destination-profile short-txt-destination alert-group syslog-group-port`	Configures the predefined destination profile (short-txt-destination) to receive Call Home Notifications corresponding to syslog messages for the port facility.
Step 4	`switch(config-callhome)# destination-profile short-txt-destination message-level 5`	Optional. Configures the predefined destination-profile (short-txt-destination) to send a Call Home message for syslog messages whose severity levels map to Call Home severity level of 5 or greater. The default is message level 0 (all syslog messages).

RMON-Based Alerts

You can configure the switch to send Call Home notifications corresponding to RMON alert triggers. All RMON-based Call Home messages have their message level set to NOTIFY (2). The RMON alert group is defined for all RMON-based Call Home alerts. To receive an RMON-based Call Home alert, you must associate a destination profile with the RMON alert group.

Configuring RMON Alerts

To configure RMON alert groups, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# destination-profile xml-destination alert-group rmon`	Optional. Configures a destination message profile (rmon_group) to send Call Home notifications for configured RMON messages.

Configuring E-Mail Options

You can configure the from, reply-to, and return-receipt e-mail addresses. While most e-mail address configurations are optional, you must configure the SMTP server address for the Call Home functionality to work.

Configuring General E-Mail Options

To configure general e-mail options, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# transport e-mail from user@company1.com`	Optional. Configures the from e-mail address.
Step 4	`switch(config-callhome)# transport e-mail reply-to person@place.com`	Optional. Configures the reply-to e-mail address to which all responses should be sent.

Configuring General E-Mail Options Using HTTPS Support

The HTTPS support for Call Home provides a transport method called HTTP. HTTPS support is used for a secure communication, and HTTP is used for nonsecure communication. You can configure an HTTP URL for the Call Home destination profile as a destination. The URL link can be from a secure server or nonsecure server. For a destination profile configured with the HTTP URL, the Call Home message is posted to the HTTP URL link.

Note The Call Home HTTP configuration can be distributed over CFS on the switches running NX-OS Release 4.2(1) and later. The Call Home HTTP configuration cannot be distributed to switches that support the nondistributable HTTP configuration. Switches running lower versions than NX-OS Release 4.2(1) and later will ignore the HTTP configuration.

Configuring HTTPS Support

Any predefined or user-defined destination profiles can be configured with the HTTPS URL address.

To configure the HTTPS URL address for any destination profile, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.

Step 3	`switch(config-callhome)#`
	`destination-profile`
	`full-txt-destination http`
	`https://httpssever.com/Service`
		Optional. Configures the predefined full-txt-destination profile with a HTTPS URL address.
		The Call Home message in full-txt format is uploaded at the configured HTTPS URL address.

Step 4	`switch(config-callhome)#`
	`destination-profile CiscoTAC-1 http`
	`https://httpssever.com/Service`
		Optional. Configures the predefined CiscoTAC-1 profile with a HTTPS URL address.
		The Call Home messge in XML format is uploaded at the configured HTTPS URL address.

Step 5	`switch(config-callhome)#`
	`destination-profile test1 http`
	`https://httpssever.com/Service`
		Optional. Configures the user-defined destination profile with a HTTPS URL address.
		The Call Home messge in the configured format is uploaded at the configured HTTPS URL address.

Any predefined or user-defined destination profiles can be configured to enable or disable a particular transport method. The transport methods are HTTP and e-mail.

To enable or disable transport method for a destination profile, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.

Step 3	`switch(config-callhome)#`
	`destination-profile CiscoTAC-1`
	`transport-method  http`
		Optional. Enables predefined destination profile CiscoTAC-1 for http transport-method.
		Note For user-defined destination profiles, e-mail is the default. You can enable either or both transport mechanisms. If you disable both methods, e-mail will be enabled.

	Command	Purpose
Step 4	`switch(config-callhome)# no destination-profile CiscoTAC-1 transport-method email`	Optional. Disables predefined destination profile CiscoTAC-1 for e-mail transport-method.
Step 5	`switch(config-callhome)# destination-profile full-txt transport-method http`	Optional. Enables predefined full-txt-destination profile for HTTP transport method.

Configuring SMTP Server and Ports

To configure the SMTP server and port, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# transport email smtp-server 192.168.1.1`	Configures the DNS, IPv4 address, or IPv6 address of the SMTP server to reach the server. The port usage defaults to 25 if no port is specified. **Note** The port number is optional and, if required, may be changed depending on the server location.
	`switch(config-callhome)# transport email smtp-server 192.168.1.1 port 30`	

Multiple SMTP Server Support

Cisco MDS NX-OS and Cisco NX-OS 5000 Series switches support multiple SMTP servers for Call Home. Each SMTP server has a priority configured between 1 and 100, with 1 being the highest priority and 100 being the lowest. If the priority is not specified, a default value of 50 is used.

You can configure up to five SMTP servers for Call Home. The servers are contacted based on their priority. The highest priority server is contacted first. If the message fails to be sent, the next server in the list is contacted until the limit is exhausted. If two servers have equal priority, the one that was configured earlier is contacted.

If a high-priority SMTP server fails, the other servers will be contacted. A time delay may occur while sending a message. The delay is minimal if the attempt to send the message through the first SMTP server is successful. The delay may increase depending on the number of unsuccessful attempts with different SMTP servers.

Note The new configuration process is not related to the old configuration. However, if the SMTP servers are configured using both the old and new schemes, the older configuration is of the highest priority.

Multiple SMTP servers can be configured on any MDS 9000 Family switch, Cisco Nexus 5000 Series switches, and Cisco Nexus 7000 Series switches running Release 5.0(1a) or later.

The new configuration will only be distributed to switches that have multiple SMTP servers. The older switches in the fabric will ignore the new configuration received over CFS.

In a mixed fabric that has CFS enabled, the switches running NX-OS Release 5.0 can configure new functionalities and distribute the new configuration to other switches with Release 5.0 in the fabric over CFS. However, if an existing switch running NX-OS Release 4.x upgrades to Release 5.0, the new configurations will not be distributed to that switch as a CFS merge is not triggered on an upgrade. There are two options to upgrade:

- Apply new configuration only when all the switches in the fabric support them. (Recommanded option).
- Do an empty commit from an existing NX-OS Release 5.0 switch which has the new configuration.

To distribute the SMTP server configuration to devices running software releases prior to NX-OS Release 5.0 and earlier, use the following command:

```
switch(config-callhome)# transport email smtp-server
```

For multiple SMTP server capability, use the following command:

```
switch(config-callhome)# [no] transport email mail-server {ipv4 | IPV6 | hostname} [port
port number] [priority priority number]
```

Example 4-1 shows how to configure multiple SMTP servers for Call Home messages:

```
switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# callhome
switch(config-callhome)# transport email mail-server 192.0.2.10 priority 4
switch(config-callhome)# transport email mail-server 172.21.34.193
switch(config-callhome)# transport email smtp-server 10.1.1.174
switch(config-callhome)# transport email mail-server 64.72.101.213 priority 60
switch(config-callhome)# transport email from person@company.com
switch(config-callhome)# transport email reply-to person@company.com
```

Based on the configuration above, the SMTP servers would be contacted in this order:

10.1.1.174 (priority 0)

192.0.2.10 (priority 4)

172.21.34.193 (priority 50 - default)

64.72.101.213 (priority 60)

The **transport email mail-server** command is distributed only to devices running NX-OS Release 5.0(1a) or later. The **transport email smtp-server** command is distributed only to devices running earlier software releases.

Verifying Callhome Transport

The **show callhome transport** displays all of the transport-related configurations for Call Home.

```
switch# show callhome transport
from email addr:switch-mds@cisco.com
reply to email addr:someone@cisco.com

smtp server:72.163.129.201
smtp server port:1
smtp server priority:0

smtp server:10.64.74.94
smtp server port:25
smtp server priority:4
```

```
smtp server:192.168.1.10
smtp server port:25
smtp server priority:50

smtp server:mail-server-1.cisco.com
smtp server port:25
smtp server priority:100
switch#
```

The following example shows how to configure SMTP server port:

```
switch# callhome
switch(config-callhome)# transport email mail-server 192.168.10.23 port 4
switch# config t
```

The following example shows how to configure SMTP server priority:

```
switch(config-callhome)# transport email mail-server 192.168.10.23 priority 60
switch# config t
```

Periodic Inventory Notification

You can configure the switch to periodically send a message with an inventory of all software services currently enabled and running on the switch along with hardware inventory information. The inventory is modified each time the switch is restarted nondisruptively.

When you enable this feature without configuring an interval value, the Call Home message is sent every 7 days. This value ranges from 1 to 30 days. By default, this feature is disabled in all switches in the Cisco MDS 9000 Family switches and Cisco Nexus 5000 Series switches.

Enabling Periodic Inventory Notifications

To enable periodic inventory notification in a Cisco MDS 9000 Family switch or a Cisco Nexus 5000 Series switch, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters the Call Home configuration submode.
Step 3	`switch(config-callhome)# periodic-inventory notification`	Enables the periodic inventory notification feature. By default, the Call Home message is sent every 7 days.
	`switch(config-callhome)# no periodic-inventory notification`	Disables the periodic inventory notification feature (default).
Step 4	`switch(config-callhome)# periodic-inventory notification interval 15`	Configures the periodic inventory notification message to be sent every 15 days. This value ranges from 1 to 30 days.
	`switch(config-callhome)# no periodic-inventory notification interval 15`	Defaults to using the factory default of sending a Call Home message every 7 days.

Duplicate Message Throttle

You can configure a throttling mechanism to limit the number of Call Home messages received for the same event. If the same message is sent multiple times from the switch within a short period of time, you may be swamped with a large number of duplicate messages.

By default, this feature is enabled in all switches in the Cisco MDS 9000 Family switches and the Cisco Nexus 5000 Series switches. When enabled, if the number of messages sent exceeds the maximum limit of 30 messages within the 2-hour time frame, then additional messages for that alert type are discarded within that time frame. You cannot modify the time frame or the message counter limit.

If 2 hours have elapsed since the first such message was sent and a new message has to be sent, then the new message is sent and the time frame is reset to the time when the new message was sent and the count is reset to 1.

Enabling Message Throttling

To enable message throttling in a Cisco MDS 9000 Family switch or a Cisco Nexus 5000 Series switch, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters the Call Home configuration submode.
Step 3	`switch(config-callhome)# no duplicate-message throttle`	Disables the duplicate message throttling feature.
	`switch(config-callhome)# duplicate-message throttle`	Enables the duplicate message throttling feature (default).

Enabling Call Home Function

Once you have configured the contact information, you must enable the Call Home function.

To enable the Call Home function, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.

Step 3	`switch(config-callhome)# enable`
	`callhome enabled successfully`
	`switch(config-callhome)#`
		Enables the Call Home function.

	`switch(config-callhome)# disable` `switch(config-callhome)#`	Disables the Call Home function. When you disable the Call Home function, all input events are ignored. **Note** Even if Call Home is disabled, basic information for each Call Home event is sent.

Call Home Configuration Distribution

You can enable fabric distribution for all Cisco MDS 9000 Family switch and Cisco Nexus NX-OS 5000 Series switches in the fabric. When you perform Call Home configurations, and distribution is enabled, that configuration is distributed to all the switches in the fabric. However, the switch priority and the Syscontact names are not distributed.

You automatically acquire a fabric-wide lock when you enter the first configuration command operation after you enable distribution in a switch. The Call Home application uses the effective and pending database model to store or commit the configuration changes. When you commit the configuration changes, the effective database is overwritten by the configuration changes in the pending database and all the switches in the fabric receive the same configuration. After making the configuration changes, you can choose to discard the changes by aborting the changes instead of committing them. In either case, the lock is released. See Chapter 2, “Using the CFS Infrastructure” for more information on the CFS application.

Note The switch priority and the Syscontact name are not distributed.

Enabling Call Home Fabric Distribution

To enable Call Home fabric distribution, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# distribute`	Enables Call Home configuration distribution to all switches in the fabric. Acquires a fabric lock and stores all future configuration changes in the pending database.
	`switch(config-callhome)# no distribute`	Disables (default) Call Home configuration distribution to all switches in the fabric.

To commit the Call Home configuration changes, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# commit`	Distributes the configuration changes to all switches in the fabric and releases the lock. Overwrites the effective database with the changes made to the pending database.

To discard the Call Home configuration changes, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# callhome` `switch(config-callhome)#`	Enters Call Home configuration submode.
Step 3	`switch(config-callhome)# abort`	Discards the configuration changes in the pending database and releases the fabric lock.

Fabric Lock Override

If you have performed a Call Home task and have forgotten to release the lock by either committing or discarding the changes, an administrator can release the lock from any switch in the fabric. If the administrator performs this task, your changes to the pending database are discarded and the fabric lock is released.

Tip The changes are only available in the volatile directory and are subject to being discarded if the switch is restarted.

To use administrative privileges and release a locked Call Home session, use the **clear callhome session** command.

```
switch# clear callhome session
```

To use administrative privileges and release a locked Call Home session, use the **clear callhome session** command.

```
switch# clear callhome session
```

Database Merge Guidelines

See the "CFS Merge Support" section on page 2-9 for detailed concepts.

When merging two Call Home databases, follow these guidelines:

- Be aware that the merged database contains the following information:
  - A superset of all the destination profiles from the dominant and subordinate switches that take part in the merge protocol.
  - The e-mail addresses and alert groups for the destination profiles.
  - Other configuration information (for example, message throttling, periodic inventory) from the switch that existed in the dominant switch before the merge.
- Verify that two destination profiles do not have the same name (even if they have different configuration information) on the subordinate and dominant switches. If they do contain the same name, the merge operation will fail. You must then modify or delete the conflicting destination profile on the required switch.

Call Home Communications Test

You can test Call Home communications by sending a test message to the configured destination(s) or sending a test inventory message to the configured destination(s).

Testing Call Home

Use the **test** command to simulate a message generation.

To test the Call Home function, follow these steps:

	Command	Purpose

Step 1	`switch# callhome test`
	`trying to send test callhome message`
	`successfully sent test callhome message`
		Sends a test message to the configured destination(s).

Step 2	`switch# callhome test inventory`
	`trying to send test callhome message`
	`successfully sent test callhome message`
		Sends a test inventory message to the configured destination(s).

Displaying Call Home Information

Use the **show callhome** command to display the configured Call Home information (see Examples 4-1 to 4-7).

Example 4-1 Displays Configured Call Home Information

```
switch# show callhome
callhome enabled
Callhome Information:
contact person name:who@where
contact person's e-mail:person@place.com
contact person's phone number:310-408-4000
street addr:1234 Picaboo Street, Any city, Any state, 12345
site id:Site1ManhattanNewYork
customer id:Customer1234
contract id:Cisco1234
switch priority:0
```

Example 4-2 Displays Information for All Destination Profiles (Predefined and User-Defined)

```
switch# show callhome destination-profile
XML destination profile information
maximum message size:500000
message format:XML
message-level:0
e-mail addresses configured:
alert groups configured:
cisco_tac

test destination profile information
maximum message size:100000
message format:full-txt
message-level:5
e-mail addresses configured:
```

```
admin@yourcompany.com

alert groups configured:
test

full-txt destination profile information
maximum message size:500000
message format:full-txt
message-level:0
e-mail addresses configured:

alert groups configured:
all

short-txt destination profile information
maximum message size:4000
message format:short-txt
message-level:0
e-mail addresses configured:

alert groups configured:
all
```

Example 4-3 Displays Information for a User-defined Destination Profile

```
switch# show callhome destination-profile test
test destination profile information
maximum message size:100000
message format:full-txt
message-level:5
e-mail addresses configured:
user@company.com
alert groups configured:
test
```

Example 4-4 Displays the Full-Text Profile

```
switch# show callhome destination-profile profile full-txt-destination
full-txt destination profile information
maximum message size:250000
e-mail addresses configured:
person2@company2.com
```

Example 4-5 Displays the Short-Text Profile

```
switch# show callhome destination-profile profile short-txt-destination
Short-txt destination profile information
maximum message size:4000
e-mail addresses configured:
person2@company2.com
```

Example 4-6 Displays the XML Destination Profile

```
switch# show callhome destination-profile profile XML-destination
XML destination profile information
maximum message size:250000
e-mail addresses configured:
findout@.cisco.com
```

Example 4-7 Displays E-Mail and SMTP Information

```
switch# show callhome transport-e-mail
from e-mail addr:user@company1.com
reply to e-mail addr:pointer@company.com
return receipt e-mail addr:user@company1.com
smtp server:server.company.com
smtp server port:25
```

Clearing Call Home Name Server Database

When the Call Home name server database is full, a new entry cannot be added. The device is not allowed to come online.

To clear the name server database, increase the database size or perform a cleanup by removing unused devices. A total of 20,000 name server entries are supported.

Verifying the Number of Name Server Database Entries

To verify the number of name server database entries, follow these steps:

	Command	Purpose
Step 1	`switch# show fcns internal info global`	Displays the number of device entries in the name server database.
Step 2	`switch# show fcns internal info`	Displays the number of devices in the name server database at the end of the output.

Configuring EMC E-mail Home Delayed Traps

Fabric Manager can be configured to generate EMC E-mail Home XML e-mail messages. In SAN-OS Release 3.x or earlier, Fabric Manager listens to interface traps and generates EMC E-mail Home e-mail messages. Link traps are generated when an interface goes to down from up or vice versa. For example, if there is a scheduled server reboot, the link goes down and Fabric Manager generates an e-mail notification.

Cisco NX-OS Release 4.1(3) provides the ability to generate a delayed trap so that the number of generated e-mail messages is reduced. This method filters server reboots and avoids generating unnecessary EMC E-mail Home e-mail messages. In NX-OS Release 4.1(3), users have the ability to select the current existing feature or this new delayed trap feature.

Configuring Delayed Traps

To enable the delayed trap feature, perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.

	Command	Purpose
Step 2	`switch(config)# system delayed-traps enable mode FX`	Enables the system-delayed trap feature.
Step 3	`switch(config)# system delayed-traps timer <1-60>`	Configures the system-delayed trap timeout value. If no value is entered, a default value of 4 minutes is used. You can choose any value between 1 to 60 minutes.

To disable the delayed trap feature, perform this task:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# no system delayed-traps enable mode FX`	Disables the system-delayed trap feature. This command is used only for the F/FL operationally UP ports. This feature does not apply to E/TE links. By default, this feature is disabled. You have to explicitly enable this feature. Enabling the feature will not affect the existing link-level traps.

Displaying Delayed Traps Information

Use the **show running-config | in delay** CLI command to display the system-delayed trap state as shown in Example 4-8 and Example 4-9. If no timer value is specified or if the timer value is set to 4 minutes, the following is displayed:

Example 4-8 Displays the Delayed Trap Information with No Timer Value (Set to the Default 4 Minutes)

```
switch# show running-config | in delay
system delayed-traps enable mode FX
switch#
```

If the timer value is set to any other value other than 4 minutes, the following is displayed:

Example 4-9 Displays the Delayed Trap Information with a Timer Value Other Than 4 Minutes

```
switch# show running-config | in delay
system delayed-traps enable mode FX
system delayed-traps timer 5
switch#
```

Sample Syslog Alert Notification in Full-txt Format

```
source:MDS9000
Switch Priority:7
Device Id:DS-C9506@C@FG@07120011
Customer Id:basu
Contract Id:123
Site Id:San Jose
```

```
Server Id:DS-C9506@C@FG@07120011
Time of Event:2004-10-08T11:10:44
Message Name:SYSLOG_ALERT
Message Type:Syslog
Severity Level:2
System Name:10.76.100.177
Contact Name:Basavaraj B
Contact e-mail:admin@yourcompany.com
Contact Phone:+91-80-310-1718
Street Address:#71 , Miller's Road
Event Description:2004 Oct  8 11:10:44 10.76.100.177 %PORT-5-IF_TRUNK_UP: %$VSAN 1%$
Interface fc2/5, vsan 1 is up

syslog_facility:PORT
start chassis information:
Affected Chassis:DS-C9506
Affected Chassis Serial Number:FG@07120011
Affected Chassis Hardware Version:0.104
Affected Chassis Software Version:3.1(1)
Affected Chassis Part No:73-8607-01
end chassis information:
```

Sample Syslog Alert Notification in XML Format

```
<?xml version="1.0" encoding="UTF-8" ?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.cisco.com/2004/01/aml-session"
soap-env:mustUnderstand="true"
soap-env:role="http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.cisco.com/neddce/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
<aml-session:MessageId>1004:FOX090306QT:3E55A81A</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.cisco.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2003-02-21 04:16:18 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>MDS</aml-block:Name>
<aml-block:Version>4.1</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>1005:FOX090306QT:3E55A81A</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>6</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
```

```
<ch:EventTime>2003-02-21 04:16:18 GMT+00:00</ch:EventTime>
<ch:MessageDescription>LICENSE_VIOLATION 2003 Feb 21 04:16:18 switch %$
%DAEMON-3-SYSTEM_MSG: &lt;&lt;%LICMGR-3-LOG_LICAPP_NO_LIC&gt;&gt; License file is missing
for feature SAN_EXTN_OVER_IP</ch:MessageDescription>
<ch:Event>
<ch:Type>syslog</ch:Type>
<ch:SubType>LICENSE_VIOLATION</ch:SubType>
<ch:Brand>Cisco</ch:Brand>
<ch:Series>MDS9000</ch:Series>
</ch:Event>
<ch:CustomerData>
<ch:UserData>
<ch:e-mail>esajjana@cisco.com</ch:e-mail>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId>eeranna</ch:CustomerId>
<ch:SiteId>Bangalore</ch:SiteId>
<ch:ContractId>123</ch:ContractId>
<ch:DeviceId>DS-C9216I-K9@C@FOX090306QT</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>switch</ch:Name>
<ch:Contact>Eeranna</ch:Contact>
<ch:Contacte-mail>esajjana@cisco.com</ch:Contacte-mail>
<ch:ContactPhoneNumber>+91-80-310-1718</ch:ContactPhoneNumber>
<ch:StreetAddress>#71, Miller&apos;s Road</ch:StreetAddress> </ch:SystemInfo>
</ch:CustomerData> <ch:Device> <rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.0">
<rme:Model>DS-C9216I-K9</rme:Model>
<rme:HardwareVersion>1.0</rme:HardwareVersion>
<rme:SerialNumber>FOX090306QT</rme:SerialNumber>
</rme:Chassis>
</ch:Device>
</ch:CallHome>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging logfile | tail -n 200</aml-block:Name> <aml-block:Data
encoding="plain">
<![CDATA[syslog_show:: command: 1055 param_count: 0
2003 Feb 21 04:11:48  %KERN-2-SYSTEM_MSG: Starting kernel... - kernel
2003 Feb 21 04:11:48  %KERN-3-SYSTEM_MSG: CMOS: Module initialized - kernel
2003 Feb 21 04:11:48  %KERN-2-SYSTEM_MSG: CARD TYPE: KING BB Index = 2344 - kernel
2003 Feb 21 04:12:04  %MODULE-5-ACTIVE_SUP_OK: Supervisor 1 is active (serial:
JAB100700MC)
2003 Feb 21 04:12:04  %PLATFORM-5-MOD_STATUS: Module 1 current-status is
MOD_STATUS_ONLINE/OK
2003 Feb 21 04:12:06  %IMAGE_DNLD-SLOT1-5-ADDON_IMG_DNLD_COMPLETE:  Addon module image
download process completed. Addon Image download completed, installing image please wait..
2003 Feb 21 04:12:07  %IMAGE_DNLD-SLOT1-5-ADDON_IMG_DNLD_SUCCESSFUL:  Addon module image
download and install process successful. Addon image installed.
2003 Feb 21 04:12:08  %KERN-3-SYSTEM_MSG: klm_af_xipc: Unknown parameter `start&apos; -
kernel
2003 Feb 21 04:12:08  %KERN-3-SYSTEM_MSG: klm_ips_portcfg: Unknown parameter `start&apos;
- kernel
2003 Feb 21 04:12:08  %KERN-3-SYSTEM_MSG: klm_flamingo: Unknown parameter `start&apos; -
kernel
2003 Feb 21 04:12:10  %PORT-5-IF_UP: Interface mgmt0 is up
2003 Feb 21 04:12:21 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature ENTERPRISE_PKG.
2003 Feb 21 04:12:21 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature SAN_EXTN_OVER_IP.
2003 Feb 21 04:12:21 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature ENTERPRISE_PKG.
```

```
2003 Feb 21 04:12:21 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature SAN_EXTN_OVER_IP.
2003 Feb 21 04:12:23 switch %PLATFORM-5-MOD_STATUS: Module 1 current-status is
MOD_STATUS_ONLINE/OK
2003 Feb 21 04:12:23 switch %MODULE-5-MOD_OK: Module 1 is online (serial: JAB100700MC)
2003 Feb 21 04:12:25 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/1 is down
(Administratively down)
2003 Feb 21 04:12:25 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/2 is down
(Administratively down)
2003 Feb 21 04:12:25 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/3 is down
(Administratively down)
2003 Feb 21 04:12:25 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/4 is down
(Administratively down)
2003 Feb 21 04:12:26 switch %PLATFORM-5-PS_STATUS: PowerSupply 1 current-status is PS_FAIL
2003 Feb 21 04:12:26 switch %PLATFORM-2-PS_FAIL: Power supply 1 failed or shut down
(Serial number QCS1007109F)
2003 Feb 21 04:12:26 switch %PLATFORM-5-PS_FOUND: Power supply 2 found (Serial number
QCS1007109R)
2003 Feb 21 04:12:26 switch %PLATFORM-2-PS_OK: Power supply 2 ok (Serial number
QCS1007109R)
2003 Feb 21 04:12:26 switch %PLATFORM-5-PS_STATUS: PowerSupply 2 current-status is PS_OK
2003 Feb 21 04:12:26 switch %PLATFORM-2-PS_FANOK: Fan in Power supply 2 ok
2003 Feb 21 04:12:26 switch %PLATFORM-5-FAN_DETECT: Fan module 1 (Serial number
NWG0901031X) ChassisFan1 detected
2003 Feb 21 04:12:26 switch %PLATFORM-2-FAN_OK: Fan module ok
2003 Feb 21 04:12:26 switch %PLATFORM-2-CHASSIS_CLKMODOK: Chassis clock module A ok
2003 Feb 21 04:12:26 switch %PLATFORM-2-CHASSIS_CLKSRC: Current chassis clock source is
clock-A
2003 Feb 21 04:12:26 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/5 is down
(Administratively down)
2003 Feb 21 04:12:26 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/6 is down
(Administratively down)
2003 Feb 21 04:12:26 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/7 is down
(Administratively down)
2003 Feb 21 04:12:26 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/8 is down
(Administratively down)
2003 Feb 21 04:12:26 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/9 is down
(Administratively down)
2003 Feb 21 04:12:26 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/10 is
down (Administratively down)
2003 Feb 21 04:12:27 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/11 is
down (Administratively down)
2003 Feb 21 04:12:27 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/12 is
down (Administratively down)
2003 Feb 21 04:12:27 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/13 is
down (Administratively down)
2003 Feb 21 04:12:27 switch %PORT-5-IF_DOWN_ADMIN_DOWN: %$VSAN 1%$ Interface fc1/14 is
down (Administratively down)
2003 Feb 21 04:12:30 switch %PLATFORM-2-MOD_DETECT: Module 2 detected (Serial number
JAB0923016X) Module-Type IP Storage Services Module Model DS-X9304-SMIP
2003 Feb 21 04:12:30 switch %MODULE-2-MOD_UNKNOWN: Module type [25] in slot 2 is not
supported
2003 Feb 21 04:12:45 switch %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configured from vty by root on
console0
2003 Feb 21 04:14:06 switch %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configured from vty by admin on
console0
2003 Feb 21 04:15:12 switch %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configured from vty by admin on
console0
2003 Feb 21 04:15:52 switch %SYSMGR-3-BASIC_TRACE: core_copy: PID 1643 with message Core
not generated by system for licmgr(0). WCOREDUMP(9) returned zero .
2003 Feb 21 04:15:52 switch %SYSMGR-2-SERVICE_CRASHED: Service \"licmgr\" (PID 2272)
hasn&amp;apos;t caught signal 9 (no core).
2003 Feb 21 04:16:18 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature ENTERPRISE_PKG.
```

```
2003 Feb 21 04:16:18 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature SAN_EXTN_OVER_IP.
2003 Feb 21 04:16:18 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature ENTERPRISE_PKG.
2003 Feb 21 04:16:18 switch %LICMGR-3-LOG_LIC_FILE_MISSING: License file(s) missing for
feature SAN_EXTN_OVER_IP.
2003 Feb 21 04:16:18 switch %CALLHOME-2-EVENT: LICENSE_VIOLATION
2003 Feb 21 04:16:18 switch %CALLHOME-2-EVENT: LICENSE_VIOLATION
2003 Feb 21 04:16:18 switch %CALLHOME-2-EVENT: LICENSE_VIOLATION
2003 Feb 21 04:16:18 switch %CALLHOME-2-EVENT: LICENSE_VIOLATION ]]> </aml-block:Data>
</aml-block:Attachment> <aml-block:Attachment type="inline"> <aml-block:Name>show license
usage</aml-block:Name> <aml-block:Data encoding="plain">
<![CDATA[Feature                        Ins  Lic   Status Expiry Date Comments
                                  Count
--------------------------------------------------------------------------------
DMM_184_PKG                    No    0   Unused             Grace expired
FM_SERVER_PKG                  No    -   Unused             Grace expired
MAINFRAME_PKG                  No    -   Unused             Grace expired
ENTERPRISE_PKG                 Yes   -   Unused never       license missing
DMM_FOR_SSM_PKG                No    0   Unused             Grace expired
SAN_EXTN_OVER_IP               Yes   8   Unused never       8 license(s) missing
PORT_ACTIVATION_PKG            No    0   Unused             -
SME_FOR_IPS_184_PKG            No    0   Unused             Grace expired
STORAGE_SERVICES_184           No    0   Unused             Grace expired
SAN_EXTN_OVER_IP_18_4          No    0   Unused             Grace expired
SAN_EXTN_OVER_IP_IPS2          No    0   Unused             Grace expired
SAN_EXTN_OVER_IP_IPS4          No    0   Unused             Grace expired
STORAGE_SERVICES_SSN16         No    0   Unused             Grace expired
10G_PORT_ACTIVATION_PKG        No    0   Unused             -
STORAGE_SERVICES_ENABLER_PKG   No    0   Unused             Grace expired
--------------------------------------------------------------------------------
**** WARNING: License file(s) missing. **** ]]> </aml-block:Data> </aml-block:Attachment>
</aml-block:Attachments> </aml-block:Block> </soap-env:Body> </soap-env:Envelope>
```

Sample RMON Notification in XML Format

```
<?xml version="1.0" encoding="UTF-8" ?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.cisco.com/2004/01/aml-session"
soap-env:mustUnderstand="true"
soap-env:role="http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.cisco.com/neddce/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
<aml-session:MessageId>1086:FHH0927006V:48BA26BD</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.cisco.com/2005/05/callhome/diagnostic</aml-block:Type>
<aml-block:CreationDate>2008-08-31 05:06:05 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>MDS</aml-block:Name>
<aml-block:Version>4.1</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>1087:FHH0927006V:48BA26BD</aml-block:GroupId>
```

```
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>2</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
<ch:EventTime>2008-08-31 05:06:05 GMT+00:00</ch:EventTime>
<ch:MessageDescription>RMON_ALERT WARNING(4) Falling:iso.3.6.1.4.1.9.9.305.1.1.1.0=1 &lt;=
89:1, 4</ch:MessageDescription>
<ch:Event>
<ch:Type>diagnostic</ch:Type>
<ch:SubType>GOLD-major</ch:SubType>
<ch:Brand>Cisco</ch:Brand>
<ch:Series>MDS9000</ch:Series>
</ch:Event>
<ch:CustomerData>
<ch:UserData>
<ch:e-mail>mchinn@cisco.com</ch:e-mail>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId>12ss</ch:CustomerId>
<ch:SiteId>2233</ch:SiteId>
<ch:ContractId>rrr55</ch:ContractId>
<ch:DeviceId>DS-C9513@C@FHH0927006V</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>sw172-22-46-174</ch:Name>
<ch:Contact>Mani</ch:Contact>
<ch:Contacte-mail>mchinn@cisco.com</ch:Contacte-mail>
<ch:ContactPhoneNumber>+1-800-304-1234</ch:ContactPhoneNumber>
<ch:StreetAddress>1234 wwee</ch:StreetAddress>
</ch:SystemInfo>
</ch:CustomerData>
<ch:Device>
<rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.0">
<rme:Model>DS-C9513</rme:Model>
<rme:HardwareVersion>0.205</rme:HardwareVersion>
<rme:SerialNumber>FHH0927006V</rme:SerialNumber>
</rme:Chassis>
</ch:Device>
</ch:CallHome>
</aml-block:Content>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```

Event Triggers

This section discusses Call Home trigger events. Trigger events are divided into categories, with each category assigned CLI commands to execute when the event occurs. The command output is included in the transmitted message. Table 4-2 lists the trigger events.

Table 4-2 Event Triggers

Event	Alert Group	Event Name	Description	Call Home Message Level
Call Home	System and CISCO_TAC	SW_CRASH	A software process has crashed with a stateless restart, indicating an interruption of a service.	5
	System and CISCO_TAC	SW_SYSTEM_INCONSISTENT	Inconsistency detected in software or file system.	5
	Environmental and CISCO_TAC	TEMPERATURE_ALARM	Thermal sensor indicates temperature reached operating threshold.	6
		POWER_SUPPLY_FAILURE	Power supply failed.	6
		FAN_FAILURE	Cooling fan has failed.	5
	Line Card Hardware and CISCO_TAC	LINECARD_FAILURE	Line card hardware operation failed.	7
		POWER_UP_DIAGNOSTICS_FAILURE	Line card hardware failed power-up diagnostics.	7
	Line Card Hardware and CISCO_TAC	PORT_FAILURE	Hardware failure of interface port(s).	6
	Line Card Hardware, Supervisor Hardware, and CISCO_TAC	BOOTFLASH_FAILURE	Failure of boot compact flash card.	6
	Supervisor Hardware and CISCO_TAC	NVRAM_FAILURE	Hardware failure of NVRAM on supervisor hardware.	6
	Supervisor Hardware and CISCO_TAC	FREEDISK_FAILURE	Free disk space is below a threshold on supervisor hardware.	6
	Supervisor Hardware and CISCO_TAC	SUP_FAILURE	Supervisor hardware operation failed.	7
		POWER_UP_DIAGNOSTICS_FAILURE	Supervisor hardware failed power-up diagnostics.	7
	Supervisor Hardware and CISCO_TAC	INBAND_FAILURE	Failure of in-band communications path.	7
	Supervisor Hardware and CISCO_TAC	EOBC_FAILURE	Ethernet out-of-band channel communications failure.	6

Table 4-2 Event Triggers (continued)

Event	Alert Group	Event Name	Description	Call Home Message Level
Call Home	Supervisor Hardware and CISCO_TAC	MGMT_PORT_FAILURE	Hardware failure of management Ethernet port.	5
	License	LICENSE_VIOLATION	Feature in use is not licensed, and are turned off after grace period expiration.	6
Inventory	Inventory and CISCO_TAC	COLD_BOOT	Switch is powered up and reset to a cold boot sequence.	2
		HARDWARE_INSERTION	New piece of hardware inserted into the chassis.	2
		HARDWARE_REMOVAL	Hardware removed from the chassis.	2
Test	Test and CISCO_TAC	TEST	User generated test.	2
Port syslog	Syslog-group-port	SYSLOG_ALERT	Syslog messages corresponding to the port facility.	2
RMON	RMON	RMON_ALERT	RMON alert trigger messages.	2

Table 4-3 lists event categories and command outputs.

Table 4-3 Event Categories and Executed Commands

Event Category	Description	Executed Commands
System show module show version show tech-support platform show tech-support sysmgr show hardware show sprom all	Events generated by failure of a software system that is critical to unit operation.	**show tech-support** **show system redundancy status**
Environmental show module show version show environment show logging logfile | tail -n 200	Events related to power, fan, and environment sensing elements such as temperature alarms.	**show module** **show environment**

Table 4-3 Event Categories and Executed Commands (continued)

Event Category	Description	Executed Commands
Line Card Hardware show module show version show tech-support platform show tech-support sysmgr show hardware show sprom all	Events related to standard or intelligent line card hardware.	**show tech-support**
Supervisor Hardware show module show version show tech-support platform show tech-support sysmgr show hardware show sprom all	Events related to supervisor modules.	**show tech-support**
Inventory show module show version show hardware show inventory show system uptime show sprom all show license usage	Inventory status is provided whenever a unit is cold booted, or when FRUs are inserted or removed. This is considered a noncritical event, and the information is used for status and entitlement.	**show version**
Test show module show version	User generated test message.	**show version**

Call Home Message Levels

Call Home messages (sent for syslog alert groups) have the syslog severity level mapped to the Call Home message level (see the "Syslog-Based Alerts" section on page 4-13).

This section discusses the severity levels for a Call Home message when using one or more switches in the Cisco MDS 9000 Family and the Cisco Nexus 5000 Series switches. Call Home message levels are preassigned per event type.

Severity levels range from 0 to 9, with 9 having the highest urgency. Each syslog level has keywords and a corresponding syslog level as listed in Table 4-4.

Note Call Home does not change the syslog message level in the message text. The syslog message texts in the Call Home log appear as they are described in the *Cisco MDS 9000 Family System Messages Reference*.

Note Call Home severity levels are not the same as system message logging severity levels (see the *Cisco MDS 9000 Family System Messages Reference*).

Table 4-4 Severity and Syslog Level Mapping

Call Home Level	Keyword Used	Syslog Level	Description
Catastrophic (9)	**Catastrophic**	N/A	Network wide catastrophic failure.
Disaster (8)	**Disaster**	N/A	Significant network impact.
Fatal (7)	**Fatal**	Emergency (0)	System is unusable.
Critical (6)	**Critical**	Alert (1)	Critical conditions, immediate attention needed.
Major (5)	**Major**	Critical (2)	Major conditions.
Minor (4)	**Minor**	Error (3)	Minor conditions.
Warning (3)	**Warning**	Warning (4)	Warning conditions.
Notify (2)	**Notification**	Notice (5)	Basic notification and informational messages. Possibly independently insignificant.
Normal (1)	**Normal**	Information (6)	Normal event signifying return to normal state.
Debug (0)	**Debugging**	Debug (7)	Debugging messages.

Message Contents

The following contact information can be configured on the switch:

- Name of the contact person
- Phone number of the contact person
- E-mail address of the contact person
- Mailing address to which replacement parts must be shipped, if required
- Site ID of the network where the site is deployed

- Contract ID to identify the service contract of the customer with the service provider

Table 4-5 describes the short text formatting option for all message types.

Table 4-5 Short Text Messages

Data Item	Description
Device identification	Configured device name
Date/time stamp	Time stamp of the triggering event
Error isolation message	Plain English description of triggering event
Alarm urgency level	Error level such as that applied to system message

Table 4-6, Table 4-7, and Table 4-8 display the information contained in plain text and XML messages.

Table 4-6 Reactive Event Message Format

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
Time stamp	Date and time stamp of event in ISO time notation: *YYYY-MM-DDTHH:MM:SS*. **Note** The time zone or daylight savings time (DST) offset from UTC has already been added or subtracted. T is the hardcoded limiter for the time.	/mml/header/time - ch:EventTime
Message name	Name of message. Specific event names are listed in the "Event Triggers" section on page 4-31.	/mml/header/name
Message type	Specifically "Call Home."	/mml/header/type - ch:Type
Message group	Specifically "reactive."	/mml/header/group
Severity level	Severity level of message (see Table 4-4).	/mml/header/level - aml-block:Severity
Source ID	Product type for routing.	/mml/header/source - ch:Series
Device ID	Unique device identifier (UDI) for end device generating message. This field should empty if the message is non-specific to a fabric switch. Format is *type***@***Sid***@***serial*, where: • *type* is the product model number from backplane SEEPROM. • **@** is a separator character. • *Sid* is "C," identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: DS-C9509@C@12345678	/mml/ header/deviceId
Customer ID	Optional user-configurable field used for contract info or other ID by any support service.	/mml/header/customerID - ch:CustomerId
Contract ID	Optional user-configurable field used for contract info or other ID by any support service.	/mml/header/contractId - ch:ContractId>
Site ID	Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service.	/mml/header/siterId - ch:SiteId

Table 4-6 Reactive Event Message Format (continued)

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
Server ID	If the message is generated from the fabric switch, it is the unique device identifier (UDI) of the switch. Format is *type*@*Sid*@*serial*, where: • *type* is the product model number from backplane SEEPROM. • @ is a separator character. • *Sid* is "C,," identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: DS-C9509@C@12345678	/mml/header/serverId - -blank-
Message description	Short text describing the error.	/mml/body/msgDesc - ch:MessageDescription
Device name	Node that experienced the event. This is the host name of the device.	/mml/body/sysName - ch:SystemInfo/Name
Contact name	Name of person to contact for issues associated with the node experiencing the event.	/mml/body/sysContact - ch:SystemInfo/Contact
Contact e-mail	E-mail address of person identified as contact for this unit.	/mml/body/sysContacte-mail - ch:SystemInfo/Contacte-mail
Contact phone number	Phone number of the person identified as the contact for this unit.	/mml/body/sysContactPhone Number - ch:SystemInfo/ContactPhone Number
Street address	Optional field containing street address for RMA part shipments associated with this unit.	/mml/body/sysStreetAddress - ch:SystemInfo/StreetAddress
Model name	Model name of the switch. This is the specific model as part of a product family name.	/mml/body/chassis/name - rme:Chassis/Model
Serial number	Chassis serial number of the unit.	/mml/body/chassis/serialNo - rme:Chassis/SerialNumber
Chassis part number	Top assembly number of the chassis.	/mml/body/fru/partNo - rme:chassis/Card/PartNumber
Chassis hardware version	Hardware version of chassis.	/mml/body/chassis/hwVersion - rme:Chassis/HardwareVersion
Supervisor module software version	Top level software version.	/mml/body/fru/swVersion - rme:chassis/Card/SoftwareIdentity
Affected FRU name	Name of the affected FRU generating the event message.	/mml/body/fru/name - rme:chassis/Card/Model
Affected FRU serial number	Serial number of affected FRU.	/mml/body/fru/serialNo - rme:chassis/Card/SerialNumber
Affected FRU part number	Part number of affected FRU.	/mml/body/fru/partNo - rme:chassis/Card/PartNumber

Table 4-6 Reactive Event Message Format (continued)

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
FRU slot	Slot number of FRU generating the event message.	/mml/body/fru/slot - rme:chassis/Card/LocationWithinContainer
FRU hardware version	Hardware version of affected FRU.	/mml/body/fru/hwVersion - rme:chassis/Card/SoftwareIdentity
FRU software version	Software version(s) running on affected FRU.	/mml/body/fru/swVersion - rme:chassis/Card/SoftwareIdentity
Command output name	The exact name of the issued command.	/mml/attachments/attachment/name - aml-block:Attachment/Name
Attachment type	Specifically command output.	/mml/attachments/attachment/type - aml-block:Attachment type
MIME type	Normally text or plain or encoding type.	/mml/attachments/attachment/mime - aml-block:Attachment/Data encoding
Command output text	Output of command automatically executed (see Table 4-3).	/mml/attachments/attachment/atdata - aml-block:Attachment/Data

Table 4-7 Inventory Event Message Format

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
Time stamp	Date and time stamp of event in ISO time notation: *YYYY-MM-DDTHH:MM:SS*. **Note** The time zone or daylight savings time (DST) offset from UTC has already been added or subtracted. T is the hardcoded limiter for the time.	/mml/header/time - ch:EventTime
Message name	Name of message. Specifically “Inventory Update” Specific event names are listed in the “Event Triggers” section on page 4-31.	/mml/header/name
Message type	Specifically “Inventory Update.”	/mml/header/type - ch-inv:Type
Message group	Specifically “proactive.”	/mml/header/group
Severity level	Severity level of inventory event is level 2 (see Table 4-4).	/mml/header/level - aml-block:Severity
Source ID	Product type for routing at Cisco. Specifically “MDS 9000.”	/mml/header/source - ch-inv:Series

Table 4-7 Inventory Event Message Format (continued)

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
Device ID	Unique Device Identifier (UDI) for end device generating message. This field should empty if the message is non-specific to a fabric switch. Format is *type***@***Sid***@***serial*, where: • *type* is the product model number from backplane SEEPROM. • **@** is a separator character. • *Sid* is "C,," identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: DS-C9509@C@12345678	/mml/ header /deviceId
Customer ID	Optional user-configurable field used for contact info or other ID by any support service.	/mml/header/customerID - ch-inv:CustomerId
Contract ID	Optional user-configurable field used for contact info or other ID by any support service.	/mml/header/contractId - ch-inv:ContractId>
Site ID	Optional user-configurable field, can be used for Cisco-supplied site ID or other data meaningful to alternate support service.	/mml/header/siterId - ch-inv:SiteId
Server ID	If the message is generated from the fabric switch, it is the Unique device identifier (UDI) of the switch. Format is *type***@***Sid***@***serial*, where: • *type* is the product model number from backplane SEEPROM. • **@** is a separator character. • *Sid* is "C,," identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: DS-C9509@C@12345678	/mml/header/serverId - -blank-
Message description	Short text describing the error.	/mml/body/msgDesc - ch-inv:MessageDescription
Device name	Node that experienced the event.	/mml/body/sysName - ch-inv:SystemInfo/Name
Contact name	Name of person to contact for issues associated with the node experiencing the event.	/mml/body/sysContact - ch-inv:SystemInfo/Contact
Contact e-mail	E-mail address of person identified as contact for this unit.	/mml/body/sysContacte-mail - ch-inv:SystemInfo/Contacte-mail
Contact phone number	Phone number of the person identified as the contact for this unit.	/mml/body/sysContactPhone Number - ch-inv:SystemInfo/ContactPhoneNumber
Street address	Optional field containing street address for RMA part shipments associated with this unit.	/mml/body/sysStreetAddress - ch-inv:SystemInfo/StreetAddress

Table 4-7 Inventory Event Message Format (continued)

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
Model name	Model name of the unit. This is the specific model as part of a product family name.	/mml/body/chassis/name - rme:Chassis/Model
Serial number	Chassis serial number of the unit.	/mml/body/chassis/serialNo - rme:Chassis/SerialNumber
Chassis part number	Top assembly number of the chassis.	/mml/body/fru/partNo - rme:chassis/Card/PartNumber
Chassis hardware version	Hardware version of chassis.	/mml/body/fru/hwVersion - rme:chassis/Card/SoftwareIdentity
Supervisor module software version	Top level software version.	/mml/body/fru/swVersion - rme:chassis/Card/SoftwareIdentity
FRU name	Name of the affected FRU generating the event message.	/mml/body/fru/name - rme:chassis/Card/Model
FRU s/n	Serial number of FRU.	/mml/body/fru/serialNo - rme:chassis/Card/SerialNumber
FRU part number	Part number of FRU.	/mml/body/fru/partNo - rme:chassis/Card/PartNumber
FRU slot	Slot number of FRU.	/mml/body/fru/slot - rme:chassis/Card/LocationWithinContainer
FRU hardware version	Hardware version of FRU.	/mml/body/fru/hwVersion - rme:chassis/Card/SoftwareIdentity
FRU software version	Software version(s) running on FRU.	/mml/body/fru/swVersion - rme:chassis/Card/SoftwareIdentity
Command output name	The exact name of the issued command.	/mml/attachments/attachment/name - aml-block:Attachment/Name
Attachment type	Specifically command output.	/mml/attachments/attachment/type - aml-block:Attachment type
MIME type	Normally text or plain or encoding type.	/mml/attachments/attachment/mime - aml-block:Attachment/Data encoding
Command output text	Output of command automatically executed after event categories (see "Event Triggers" section on page 4-31).	/mml/attachments/attachment/atdata - aml-block:Attachment/Data

Table 4-8 User-Generated Test Message Format

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
Time stamp	Date and time stamp of event in ISO time notation: *YYYY-MM-DD*T*HH:MM:SS*. **Note** The time zone or daylight savings time (DST) offset from UTC has already been added or subtracted. T is the hardcoded limiter for the time.	/mml/header/time - ch:EventTime
Message name	Name of message. Specifically test message for test type message. Specific event names listed in the "Event Triggers" section on page 4-31).	/mml/header/name
Message type	Specifically "Test Call Home."	/mml/header/type - ch:Type
Message group	This field should be ignored by the receiving Call Home processing application, but may be populated with either "proactive" or "reactive."	/mml/header/group
Severity level	Severity level of message, test Call Home message (see Table 4-4).	/mml/header/level - aml-block:Severity
Source ID	Product type for routing.	/mml/header/source - ch:Series
Device ID	Unique device identifier (UDI) for end device generating message. This field should empty if the message is nonspecific to a fabric switch. Format is *type***@***Sid***@***serial*, where: • *type* is the product model number from backplane SEEPROM. • @ is a separator character. • *Sid* is "C" identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: DS-C9509@C@12345678	/mml/ header /deviceId
Customer ID	Optional user-configurable field used for contract info or other ID by any support service.	/mml/header/customerID - ch:CustomerId
Contract ID	Optional user-configurable field used for contract info or other ID by any support service.	/mml/header/contractId - ch:ContractId
Site ID	Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service.	/mml/header/siterId - ch:SiteId
Server ID	If the message is generated from the fabric switch, it is the Unique device identifier (UDI) of the switch. Format is *type***@***Sid***@***serial*, where: • *type* is the product model number from backplane SEEPROM. • @ is a separator character. • *Sid* is "C" identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: "DS-C9509@C@12345678	/mml/header/serverId - -blank-
Message description	Short text describing the error.	/mml/body/msgDesc - ch:MessageDescription

Table 4-8 User-Generated Test Message Format (continued)

Data Item (Plain text and XML)	Description (Plain text and XML)	XML Tag (XML only)
Device name	Switch that experienced the event.	/mml/body/sysName - ch:SystemInfo/Name
Contact name	Name of person to contact for issues associated with the node experiencing the event.	/mml/body/sysContact - ch:SystemInfo/Contact
Contact e-mail	E-mail address of person identified as contact for this unit.	/mml/body/sysContacte-mail - ch:SystemInfo/Contacte-mail
Contact phone number	Phone number of the person identified as the contact for this unit.	/mml/body/sysContactPhoneNumber - ch:SystemInfo/ContactPhoneNumber
Street address	Optional field containing street address for RMA part shipments associated with this unit.	/mml/body/sysStreetAddress - ch:SystemInfo/StreetAddress
Model name	Model name of the switch. This is the specific model as part of a product family name.	/mml/body/chassis/name - rme:Chassis/Model
Serial number	Chassis serial number of the unit.	/mml/body/chassis/serialNo - rme:Chassis/SerialNumber
Chassis part number	Top assembly number of the chassis. For example, 800-xxx-xxxx.	/mml/body/fru/partNo - rme:chassis/Card/PartNumber
Command output text	Output of command automatically executed after event categories listed in Table 4-3.	/mml/attachments/attachment/atdata - aml-block:Attachment/Data
MIME type	Normally text or plain or encoding type.	/mml/attachments/attachment/mime - aml-block:Attachment/Data encoding
Attachment type	Specifically command output.	/mml/attachments/attachment/type - aml-block:Attachment type
Command output name	The exact name of the issued command.	/mml/attachments/attachment/name - aml-block:Attachment/Name

Default Settings

Table 4-9 lists the default Call Home settings.

Table 4-9 Default Call Home Default Settings

Parameters	Default
Destination message size for a message sent in full text format.	500,000
Destination message size for a message sent in XML format.	500,000
Destination message size for a message sent in short text format.	4000
DNS or IP address of the SMTP server to reach the server if no port is specified.	25
Alert group association with profile.	All
Format type.	XML
Call Home message level.	0 (zero)

CHAPTER 5

Scheduling Maintenance Jobs

The Cisco MDS command scheduler feature helps you schedule configuration and maintenance jobs in any switch in the Cisco MDS 9000 Family.You can use this feature to schedule jobs on a one-time basis or periodically.

This chapter includes the following sections:

About the Command Scheduler

The Cisco NX-OS command scheduler provides a facility to schedule a job (set of CLI commands) or multiple jobs at a specified time in the future. The job(s) can be executed once at a specified time in the future or at periodic intervals.

Note To use the command scheduler, you do not need to obtain any license.

You can use this feature to schedule zone set changes, make QoS policy changes, back up data, save the configuration and do other similar jobs.

Scheduler Terminology

The following terms are used in this chapter.

- Job—A job is a set of NX-OS CLI commands (EXEC and config mode) that are executed as defined in the schedule.
- Schedule—A schedule determines the time when the assigned jobs must be executed. Multiple jobs can be assigned to a schedule. A schedule executes in one of two modes: one-time or periodic.
- Periodic mode—A job is executed at the user-specified periodic intervals, until it is deleted by the administrator. The following types of periodic intervals are supported:
  - Daily—The job is executed once a day.
  - Weekly—The job is executed once a week.

  - Monthly—The job is executed once a month.
  - Delta—The job is executed beginning at the specified start time and thereafter at user-specified intervals (days:hours:minutes).
- One-time mode—The job is executed once at a user-specified time.

Scheduling Guidelines

Before scheduling jobs on a Cisco MDS switch, be aware of the following guidelines:

- Prior to Cisco MDS SAN-OS Release 3.0(3), only users local to the switch could perform scheduler configuration. As of Cisco MDS SAN-OS Release 3.0(3), remote users can perform job scheduling using AAA authentication.
- Be aware that the scheduled job can fail if it encounters one of the following situations when executing the job:
  - If the license has expired for a feature at the time when a job containing commands pertaining to that feature is scheduled.
  - If a feature is disabled at the time when a job containing commands pertaining to that feature is scheduled.
  - If you have removed a module from a slot and the job has commands pertaining to the interfaces for that module or slot.
- Verify that you have configured the time. The scheduler does not have any default time configured. If you create a schedule and assign job(s) and do not configure the time, that schedule is not launched.
- While defining a job, verify that no interactive or disruptive commands (for example, **copy bootflash:** *file* **ftp:** *URI*, **write erase**, and other similar commands) are specified as part of a job because the job is executed noninteractively at the scheduled time.

Configuring the Command Scheduler

To configure the command scheduler, follow these steps:

Step 1 Enable the scheduler.

Step 2 Authorize remote user access (optional).

Step 3 Define the job.

Step 4 Specify the schedule and assign jobs to the schedule.

Step 5 Specify the time for the schedule(s).

Step 6 Verify the scheduled configuration.

This section includes the following topics:

Enabling the Command Scheduler

To use the scheduling feature, you must explicitly enable this feature 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 command scheduler feature are only available when this feature is enabled on a switch. When you disable this feature, all related configurations are automatically discarded.

To enable the command scheduling feature, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# feature scheduler`	Enables the command scheduler.
	`switch(config)# no feature scheduler`	Discards the scheduler configuration and disables the command scheduler (default).

To display the command schedule status, use the **show scheduler config** command.

```
switch# show scheduler config
config terminal
  feature scheduler
  scheduler logfile size 16
end
```

Configuring Remote User Authentication

Prior to Cisco MDS SAN-OS Release 3.0(3), only users local to the switch could perform scheduler configuration. As of Cisco MDS SAN-OS Release 3.0(3), remote users can perform job scheduling using AAA authentication.

Note AAA authentication requires the clear text password of the remote user before creating and configuring command scheduler jobs.

To configure remote user authentication, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# scheduler aaa-authentication password X12y34Z56a`	Configures a clear text password for remote users.
Step 3	`switch(config)# scheduler aaa-authentication password 0 X12y34Z56a`	Configures a clear text password for remote users.
Step 4	`switch(config)# no scheduler aaa-authentication password`	Removes the clear text password for remote users.
Step 5	`switch(config)#scheduler aaa-authentication user newuser password Z98y76X54b`	Configures a clear text password for remote user newuser.
Step 6	`switch(config)#scheduler aaa-authentication user newuser password 0 Z98y76X54b`	Configures a clear text password for remote user newuser.
Step 7	`switch(config)# no scheduler aaa-authentication password user newuser`	Removes the clear text password for remote user newuser.

To display the scheduler password configuration for remote users, use the **show running-config** command.

```
switch# show running-config | include "scheduler aaa-authentication"
scheduler aaa-authentication username newuser password 7 "C98d76S54e"
```

Note The scheduler remote user passwords are always displayed in encrypted form in the **show running-config** command output. The encrypted option (**7**) in the command exists to support applying the ASCII configuration to the switch.

Defining a Job

To define a job, you must specify the job name. This action places you in the job definition (config-job) submode. In this submode, you can define the sequence of CLI commands that the job has to perform. Be sure to exit the config-job submode to complete the job definition.

Note

- Job configuration files created using MDS NX-OS or SAN-OS releases before Cisco MDS NX-OS Release 4.1(1b) are not supported. However, you can edit the job configuration file and combine the commands within a job into a single line using a semicolon (;).
- You must exit the config-job submode for the job definition to be complete.
- You cannot modify or remove a command after exiting the config-job submode. To make changes, you must explicitly delete the defined job name and then reconfigure the job with new commands.

To define a job for the command scheduler, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# scheduler job name addMemVsan99` `switch(config-job)#`	Defines a job name and enters the job definition submode

Step 3	`switch(config-job)# command1;[command2;command3;...]`
	`switch(config-job-submode)# end`
	`switch#`
	
	Example 1:
	`switch(config-job)# config terminal;vsan database;vsan 99 interface fc1/1 - 4`
	`switch(config-job-config-vsan-db)# end`
	`switch#`
		Specifies a sequence of actions for the specified job. The defined commands are checked for validity and stored for future use.
		Note Be sure you exit the config-job submode.

	Example 2: `switch(config)# scheduler job name offpeakQOS` `switch(config-job)# conf t ; qos class-map offpeakbackupcmap match-all ; match source-wwn 23:15:00:05:30:00:2a:1f ; match destination-wwn 20:01:00:05:30:00:28:df ;exit ; qos policy-map offpeakbackuppolicy ; class offpeakbackupcmap ; priority high ; exit ; exit ; qos service policy offpeakbackuppolicy vsan 1` `switch(config-job)# end` `switch#`	Provides example of scheduling a set of configuration commands.

Step 4	**exit**
	
	Example:
	`switch(config-job)# exit`
	`switch(config)#`
		Exits the job configuration mode and saves the job.

Step 5	**show scheduler job** [*name*] Example: `switch(config)# show scheduler job`	(Optional) Displays the job information.
Step 6	**copy running-config startup-config** Example: `switch(config)# copy running-config startup-config`	(Optional) Saves this configuration change.

Verifying the Job Definition

To verify the job definition, use the **show scheduler job** command.

```
switch# show scheduler job addMemVsan99
Job Name: addMemVsan99
----------------------
  config terminal
   vsan database
    vsan 99 interface fc1/1
    vsan 99 interface fc1/2
    vsan 99 interface fc1/3
```

```
vsan 99 interface fc1/4
```

Deleting a Job

To delete a job for the command scheduler, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# no scheduler job name addMemVsan99`	Deletes a defined job and all commands defined within that job.

Specifying a Schedule

After defining jobs, you can create schedules and assign jobs to the schedule. Subsequently, you can configure the time of execution. The execution can be one-time or periodic depending on your requirements. If the time for the schedule is not configured, then it will never be executed.

Specifying a Periodic Schedule

When you specify a periodic job execution, that job is executed periodically at the specified (daily, weekly, monthly, or delta) intervals.

To specify a periodic job for the command scheduler, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# scheduler schedule name weekendbackupqos` `switch(config-schedule)#`	Defines a job schedule (weekendbackup) and enters the submode for that schedule.
	`switch(config)# no scheduler schedule name weekendbackup`	Deletes the defined schedule.
Step 3	`switch(config-schedule)# job name offpeakZoning` `switch(config-schedule)# job name offpeakQOS`	Assign two jobs offpeakZoning and offpeakQOS for this schedule.
Step 4	`switch(config-schedule)# no job name addMem99`	Deletes the job assigned for this schedule.

The following examples are for reference:

`switch(config-schedule)# time daily 23:00`	Executes the specified jobs at 11 p.m. every day.
`switch(config-schedule)# time weekly Sun:23:00`	Specifies a weekly execution every Sunday at 11 p.m.
`switch(config-schedule)# time monthly 28:23:00`	Specifies a monthly execution at 11 p.m on the 28th of each month. If you specify the date as either 29, 30, or 31, the command is automatically executed on the last day of each month.

`switch(config-schedule)# time start now repeat 48:00`	Specifies a job to be executed every 48 hours beginning 2 minutes from *now* —if today is September 24, 2004, and the time is now 2:00 p.m., the command begins executing at 2 minutes past 2:00 p.m. on September 24, 2004, and continues to execute every 48 hours after that.
`switch(config-schedule)# time start 14:00 repeat 14:00:00`	If today is September 24, 2004, (Friday), this command specifies the job to be executed every alternate Friday at 2 p.m. (every 14 days).

The most significant fields in the **time** parameter are optional. If you omit the most significant fields, the values are assumed to be the same as the current time. For example, if the current time is September 24, 2004, 22:00 hours, then the commands are executed as follows:

- The **time start 23:00 repeat 4:00:00** command implies a start time of September 24, 2004, 23:00 hours.
- The **time daily 55** command implies every day at 22:55 hours.
- The **time weekly 23:00** command implies every Friday at 23:00 hours.
- The **time monthly 23:00** command implies the 24th of every month at 23:00 hours.

Note If the time interval configured for any schedule is smaller than the time taken to execute its assigned job(s), then the subsequent schedule execution occurs only after the configured interval amount of time has elapsed following the completion time of the last iteration of the schedule. For example, a schedule is executed at 1-minute intervals and a job assigned to it takes 2 minutes to complete. If the first schedule is at 22:00 hours, the job finishes at 22:02 after which the 1-minute interval is observed, and the next execution occurs at 22:03 and finishes at 22:05.

Specifying a One-Time Schedule

When you specify a one-time job execution, that job is only executed once.

To specify a one-time job for the command scheduler, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# scheduler schedule name configureVsan99` `switch(config-schedule)#`	Defines a job schedule (configureVsan99) and enters the submode for that schedule.
Step 3	`switch(config-schedule)# job name addMemVsan99`	Assigns a predefined job name (addMemVsan99) for this schedule.
Step 4	`switch(config-schedule)# time start 2004:12:14:23:00`	Specifies a one-time execution on December 14, 2004, at 11 p.m.
	`switch(config-schedule)# no time`	Deletes the time assigned for this schedule.

Verifying Scheduler Configuration

To display the scheduler configuration, use the **show scheduler config** command.

```
switch# show scheduler config
config terminal
  feature scheduler
  scheduler logfile size 512
end

config terminal
   scheduler job name addMemVsan99
     config terminal
       vsan database
       vsan 99 interface fc1/1
       vsan 99 interface fc1/2
       vsan 99 interface fc1/3
       vsan 99 interface fc1/4
end

config terminal
  scheduler schedule name configureVsan99
    time start 2004:8:10:9:52
    job name addMemVsan99
end
```

Deleting a Schedule

To delete a schedule, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# no scheduler schedule name weekendbackup`	Deletes the defined schedule.

Removing an Assigned Job

To remove an assigned job, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# scheduler schedule name weekendbackupqos` `switch(config-schedule)#`	Specifies a job schedule (weekendbackupqos) and enters the submode for that schedule.
Step 3	`switch(config-schedule)# no job name addMem99`	Removes a job (addMem99) assigned to this schedule.

Deleting a Schedule Time

To delete the schedule time, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# scheduler schedule name weekendbackupqos` `switch(config-schedule)#`	Defines a job schedule (weekendbackup) and enters the submode for that schedule.
Step 3	`switch(config-schedule)# no time`	Deletes the schedule time configuration.The schedule will not be run until the time is configured again.

Verifying the Command Scheduler Execution Status

To verify the command scheduler execution status, use the **show scheduler schedule** command.

```
switch# show scheduler schedule configureVsan99
Schedule Name       : configureVsan99
----------------------------------
User Name           : admin
Schedule Type       : Run once on Tue Aug 10 09:48:00 2004
Last Execution Time: Tue Aug 10 09:48:00 2004
-----------------------------------------------
          Job Name                  Status
-----------------------------------------------
addMemVsan99                        Success (0)
```

Execution Logs

This section describes execution logs for the command scheduler and contains the following sections:

About Execution Logs

The command scheduler maintains a log file. While you cannot modify the contents of this file, you can change the file size. This log file is a circular log that contains the output of the job executed. If the output of the job is greater than the log file, then the output stored in this file remains truncated.

You can configure the log file size to be a maximum of 1024 KB. The default size of the execution log file is 16 KB.

Configuring Execution Logs

To configure the execution log file size, follow these steps:

	Command	Purpose
Step 1	`switch# conf t` `switch(config)#`	Enters the configuration mode.
Step 2	`switch(config)# scheduler logfile size 1024`	Configures the log file to be a maximum of 1024 KB
	`switch(config)# no scheduler logfile size`	Defaults to the log size of 16 KB.

To display the execution log file configuration, use the **show scheduler config** command.

```
switch# show scheduler config
config terminal
  feature scheduler
  scheduler logfile size 1024
end
```

Displaying Execution Log File Contents

To display the execution log for all jobs executed in the system, use the **show scheduler logfile** command.

```
switch# show scheduler logfile
Job Name       : addMemVsan99              Job Status: Success (0)
Schedule Name  : configureVsan99           User Name : admin
Completion time: Tue Aug 10 09:48:00 2004
-------------------------- Job Output --------------------------
`config terminal`
`vsan database`
`vsan 99 interface fc1/1`
`vsan 99 interface fc1/2`
`vsan 99 interface fc1/3`
`vsan 99 interface fc1/4`
```

Clearing the Execution Log File Contents

To clear the contents of the scheduler execution log file, issue the **clear scheduler logfile** command in EXEC mode.

```
switch# clear scheduler logfile
```

Default Settings

Table 5-1 lists the default settings for command scheduling parameters.

Table 5-1 Default Command Scheduler Parameters

Parameters	Default
Command scheduler	Disabled.
Log file size	16 KB.

CHAPTER 6

Monitoring System Processes and Logs

This chapter provides details on monitoring the health of the switch. It includes the following sections:

Displaying System Processes

Use the **show processes** command to obtain general information about all processes (see Example 6-1 to Example 6-6).

Example 6-1 Displays System Processes

```
switch# show processes
PID    State  PC        Start_cnt    TTY   Process
-----  -----  --------  -----------  ----  -------------
  868      S  2ae4f33e            1     -  snmpd
  869      S  2acee33e            1     -  rscn
  870      S  2ac36c24            1     -  qos
  871      S  2ac44c24            1     -  port-channel
  872      S  2ac7a33e            1     -  ntp
    -     ER         -            1     -  mdog
    -     NR         -            0     -  vbuilder
```

Where:

- ProcessId = Process ID
- State = process state.

  - D = uninterruptible sleep (usually I/O).
  - R = runnable (on run queue).
  - S = sleeping.
  - T = traced or stopped.
  - Z = defunct (“zombie”) process.
- NR = not running.
- ER = should be running but currently not-running.
- PC = current program counter in hex format.
- Start_cnt = number of times a process has been started (or restarted).
- TTY = terminal that controls the process. A hyphen usually means a daemon not running on any particular TTY.
- Process Name = name Name of the process.

Example 6-2 Displays CPU Utilization Information

```
switch# show processes cpu
PID    Runtime(ms)  Invoked   uSecs  1Sec   Process
-----  -----------  --------  -----  -----  -----------
  842         3807    137001     27    0.0  sysmgr
 1112         1220     67974     17    0.0  syslogd
 1269          220     13568     16    0.0  fcfwd
 1276         2901     15419    188    0.0  zone
 1277          738     21010     35    0.0  xbar_client
 1278         1159      6789    170    0.0  wwn
 1279          515     67617      7    0.0  vsan
```

Where:

- MemAllocated = Sum of all the dynamically allocated memory that this process has received from the system, including memory that may have been returned
- Runtime CPU Time (ms) = CPU time the process has used, expressed in milliseconds.microseconds
- Invoked = number of times the process has been invoked.
- uSecs = microseconds of CPU time on average for each process invocation.
- 1Sec = CPU utilization in percentage for the last one second.

Example 6-3 Displays Process Log Information

```
switch# show processes log
Process           PID     Normal-exit  Stack-trace  Core     Log-create-time
----------------  ------  -----------  -----------  -------  ---------------
fspf              1339              N            Y        N  Jan  5 04:25
lcm               1559              N            Y        N  Jan  2 04:49
rib               1741              N            Y        N  Jan  1 06:05
```

Where:

- Normal-exit = whether or not the process exited normally.
- Stack-trace = whether or not there is a stack trace in the log.
- Core = whether or not there exists a core file.
- Log-create-time = when the log file got generated.

Example 6-4 Displays Detail Log Information About a Process

```
switch# show processes log pid 1339
Service: fspf
Description: FSPF Routing Protocol Application

Started at Sat Jan  5 03:23:44 1980 (545631 us)
Stopped at Sat Jan  5 04:25:57 1980 (819598 us)
Uptime: 1 hours 2 minutes 2 seconds

Start type: SRV_OPTION_RESTART_STATELESS (23)
Death reason: SYSMGR_DEATH_REASON_FAILURE_SIGNAL (2)
Exit code: signal 9 (no core)
CWD: /var/sysmgr/work

Virtual Memory:

    CODE      08048000 - 0809A100
    DATA      0809B100 - 0809B65C
    BRK       0809D988 - 080CD000
    STACK     7FFFFD20
    TOTAL     23764 KB

Register Set:

    EBX 00000005         ECX 7FFFF8CC         EDX 00000000
    ESI 00000000         EDI 7FFFF6CC         EBP 7FFFF95C
    EAX FFFFFDFE         XDS 8010002B         XES 0000002B
    EAX 0000008E (orig)  EIP 2ACE133E         XCS 00000023
    EFL 00000207         ESP 7FFFF654         XSS 0000002B

Stack: 1740 bytes. ESP 7FFFF654, TOP 7FFFFD20

0x7FFFF654: 00000000 00000008 00000003 08051E95 ................
0x7FFFF664: 00000005 7FFFF8CC 00000000 00000000 ................
0x7FFFF674: 7FFFF6CC 00000001 7FFFF95C 080522CD ........\...."..
0x7FFFF684: 7FFFF9A4 00000008 7FFFFC34 2AC1F18C ........4......*
```

Example 6-5 Displays All Process Log Details

```
switch# show processes log details
================================================
Service: snmpd
Description: SNMP Agent

Started at Wed Jan  9 00:14:55 1980 (597263 us)
Stopped at Fri Jan 11 10:08:36 1980 (649860 us)
Uptime: 2 days 9 hours 53 minutes 53 seconds

Start type: SRV_OPTION_RESTART_STATEFUL (24)
Death reason: SYSMGR_DEATH_REASON_FAILURE_SIGNAL (2)
Exit code: signal 6 (core dumped)
CWD: /var/sysmgr/work
Virtual Memory:

    CODE      08048000 - 0804C4A0
    DATA      0804D4A0 - 0804D770
    BRK       0804DFC4 - 0818F000
    STACK     7FFFFCE0
    TOTAL     26656 KB
...
```

Example 6-6 Displays Memory Information About Processes

```
switch# show processes memory
PID    MemAlloc  MemLimit    MemUsed     StackBase/Ptr      Process
-----  --------  ----------  ----------  -----------------  ---------------
    1    147456  0      1667072     7ffffe50/7ffff950   init
    2         0  0            0                  0/0         ksoftirqd/0
    3         0  0            0                  0/0         desched/0
    4         0  0            0                  0/0         events/0
    5         0  0            0                  0/0         khelper
```

Where:

- MemAlloc = total memory allocated by the process.
- StackBase/Ptr = process stack base and current stack pointer in hex format.

Displaying System Status

Use the **show system** command to display system-related status information (see Example 6-7 to Example 6-10).

Example 6-7 Displays Default Switch Port States

```
switch# show system default switchport
System default port state is down
System default trunk mode is on
```

Example 6-8 Displays Error Information for a Specified ID

```
switch# show system error-id 0x401D0019
Error Facility: module
Error Description: Failed to stop Linecard Async Notification.
```

Example 6-9 Displays the System Reset Information

```
switch# Show system reset-reason module 5
----- reset reason for module 5 -----
1) At 224801 usecs after Fri Nov 21 16:36:40 2003
    Reason: Reset Requested by CLI command reload
    Service:
    Version: 1.3(1)
2) At 922828 usecs after Fri Nov 21 16:02:48 2003
    Reason: Reset Requested by CLI command reload
    Service:
    Version: 1.3(1)
3) At 318034 usecs after Fri Nov 21 14:03:36 2003
    Reason: Reset Requested by CLI command reload
    Service:
    Version: 1.3(1)
4) At 255842 usecs after Wed Nov 19 00:07:49 2003
    Reason: Reset Requested by CLI command reload
    Service:
    Version: 1.3(1)
```

The **show system reset-reason** command displays the following information:

- In a Cisco MDS 9513 Director, the last four reset-reason codes for the supervisor module in slot 7 and slot 8 are displayed. If either supervisor module is absent, the reset-reason codes for that supervisor module are not displayed.
- In a Cisco MDS 9506 or Cisco MDS 9509 switch, the last four reset-reason codes for the supervisor module in slot 5 and slot 6 are displayed. If either supervisor module is absent, the reset-reason codes for that supervisor module are not displayed.
- In a Cisco MDS 9200 Series switch, the last four reset-reason codes for the supervisor module in slot 1 are displayed.
- The **show system reset-reason module** *number* command displays the last four reset-reason codes for a specific module in a given slot. If a module is absent, then the reset-reason codes for that module are not displayed.

Use the **clear system reset-reason** command to clear the reset-reason information stored in NVRAM and volatile persistent storage.

- In a Cisco MDS 9500 Series switch, this command clears the reset-reason information stored in NVRAM in the active and standby supervisor modules.
- In a Cisco MDS 9200 Series switch, this command clears the reset-reason information stored in NVRAM in the active supervisor module.

Example 6-10 Displays System Uptime

```
switch# show system uptime
Start Time: Sun Oct 13 18:09:23 2030
Up Time:    0 days, 9 hours, 46 minutes, 26 seconds
```

Use the **show system resources** command to display system-related CPU and memory statistics (see Example 6-11).

Example 6-11 Displays System-Related CPU and Memory Information

```
switch# show system resources
Load average:   1 minute: 0.43   5 minutes: 0.17   15 minutes: 0.11
Processes   :   100 total, 2 running
CPU states  :   0.0% user,   0.0% kernel,   100.0% idle
Memory usage:   1027628K total,    313424K used,    714204K free
                   3620K buffers,   22278K cache
```

Where:

- Load average—Displays the number of running processes. The average reflects the system load over the past 1, 5, and 15 minutes.
- Processes—Displays the number of processes in the system, and how many are actually running when the command is issued.
- CPU states—Displays the CPU usage percentage in user mode, kernel mode, and idle time in the last one second.
- Memory usage—Displays the total memory, used memory, free memory, memory used for buffers, and memory used for cache in KB. Buffers and cache are also included in the *used* memory statistics.

Core and Log Files

This section contains the following topics:

Displaying Core Status

Use the **show system cores** command to display the currently configured scheme for copying cores. See Examples 6-12 to 6-15.

Example 6-12 Displays the Message when Cores are Transferred to TFTP

```
switch# show system cores
Cores are transferred to tftp://171.69.21.28/ernguyen/CORE/
```

Example 6-13 Displays the Message when Cores are Transferred to the External CF

```
switch(config)# show system cores
Cores are transferred to slot0:abcd
```

Example 6-14 Displays All Cores Available for Upload from the Active Supervisor Module

```
switch# show cores
Module-num  Process-name    PID     Core-create-time
----------  ------------    ---     ----------------
5           fspf            1524    Nov 9 03:11
6           fcc             919     Nov 9 03:09
8           acltcam         285     Nov 9 03:09
8           fib             283     Nov 9 03:08
```

Example 6-15 Displays Logs on the Local System

```
switch# show processes log
Process            PID     Normal-exit  Stack  Core   Log-create-time
-----------------  ------  -----------  -----  -----  ---------------
ExceptionLog       2862              N      Y      N  Wed Aug  6 15:08:34 2003
acl                2299              N      Y      N  Tue Oct 28 02:50:01 2003
bios_daemon        2227              N      Y      N  Mon Sep 29 15:30:51 2003
capability         2373              N      Y      N  Tue Aug 19 13:30:02 2003
core-client        2262              N      Y      N  Mon Sep 29 15:30:51 2003
fcanalyzer         5623              N      Y      N  Fri Sep 26 20:45:09 2003
fcd                12996             N      Y      N  Fri Oct 17 20:35:01 2003
fcdomain           2410              N      Y      N  Thu Jun 12 09:30:58 2003
ficon              2708              N      Y      N  Wed Nov 12 18:34:02 2003
ficonstat          9640              N      Y      N  Tue Sep 30 22:55:03 2003
flogi              1300              N      Y      N  Fri Jun 20 08:52:33 2003
idehsd             2176              N      Y      N  Tue Jun 24 05:10:56 2003
lmgrd              2220              N      N      N  Mon Sep 29 15:30:51 2003
platform           2840              N      Y      N  Sat Oct 11 18:29:42 2003
port-security      3098              N      Y      N  Sun Sep 14 22:10:28 2003
port               11818             N      Y      N  Mon Nov 17 23:13:37 2003
rlir               3195              N      Y      N  Fri Jun 27 18:01:05 2003
rscn               2319              N      Y      N  Mon Sep 29 21:19:14 2003
```

```
securityd          2239            N      N      N  Thu Oct 16 18:51:39 2003
snmpd              2364            N      Y      N  Mon Nov 17 23:19:39 2003
span               2220            N      Y      N  Mon Sep 29 21:19:13 2003
syslogd            2076            N      Y      N  Sat Oct 11 18:29:40 2003
tcap               2864            N      Y      N  Wed Aug  6 15:09:04 2003
tftpd              2021            N      Y      N  Mon Sep 29 15:30:51 2003
vpm                2930            N      N      N  Mon Nov 17 19:14:33 2003
```

Saving Cores

You can save cores (from the active supervisor module, the standby supervisor module, or any switching module) to an external CompactFlash (slot 0) or to a TFTP server in one of two ways:

- On demand—Copies a single file based on the provided process ID.
- Periodically—Copies core files periodically as configured by the user.

A new scheme overwrites any previously issued scheme. For example, if you perform another core log copy task, the cores are periodically saved to the new location or file.

Tip Be sure to create any required directory before performing this task. If the directory specified by this task does not exist, the switch software logs a system message each time a copy cores is attempted.

To copy the core and log files on demand, follow this step:

	Command	Purpose
Step 1	`switch# show cores`	Displays all the core files.
Step 2	`switch# copy core:7407 slot0:coreSample`	Copies the core file with the process ID 7407 as coreSample in slot 0.
	`switch# copy core://5/1524 tftp://1.1.1.1/abcd`	Copies cores (if any) of a process with PID 1524 generated on slot 5[1] or slot 7[2] to the TFTP server at IPv4 address 1.1.1.1. **Note** You can also use IPv6 addresses to identify the TFTP server.

1. Cisco MDS 9506 or Cisco MDS 9509 switch
2. Cisco MDS 9513 Director

If the core file for the specified process ID is not available, you see the following response:

```
switch# copy core://7/123 slot0:abcd
No matching core file found

switch# copy core:133 slot0:foo
Enter module number:7
No matching core file found

switch# copy core://7/133 slot0:foo
No matching core file found
```

If two core files exist with the same process ID, only one file is copied:

```
switch# copy core:7407 slot0:foo1
2 core files found with pid 7407
Only "/isan/tmp/logs/calc_server_log.7407.tar.gz" will be copied to the destination.
```

To copy the core and log files periodically, follow these steps:

	Command	Purpose
Step 1	`switch# show system cores`	Displays all the core files.
Step 2	`switch# config t`	Enters configuration mode.
Step 3	`switch(config)# system cores slot0:coreSample`	Copies the core file (coreSample) to slot 0.
	`switch(config)# system cores tftp://1.1.1.1/abcd`	Copies the core file (abcd) in the specified directory on the TFTP server at IPv4 address 1.1.1.1. **Note** You can also use IPv6 addresses to identify the TFTP server.
	`switch(config)# no system cores`	Disables the core files copying feature.

Saving the Last Core to Bootflash

This last core dump is automatically saved to bootflash in the /mnt/pss/ partition before the switchover or reboot occurs. Three minutes after the supervisor module reboots, the saved last core is restored from the flash partition (/mnt/pss) back to its original RAM location. This restoration is a background process and is not visible to the user.

Tip The timestamp on the restored last core file displays the time when the supervisor booted up not when the last core was actually dumped. To obtain the exact time of the last core dump, check the corresponding log file with the same PID.

To view the last core information, enter the **show cores** command in EXEC mode.

To view the time of the actual last core dump, enter the **show process log** command in EXEC mode.

Clearing the Core Directory

Use the **clear cores** command to clean out the core directory. The software clears all the core files and other cores present on the active supervisor module.

```
switch# clear cores
```

First and Last Core

The first and last core feature uses the limited system resource and retains the most important core files. Generally, the first core and the most recently generated core have the information for debugging and, the first and last core feature tries to retain the first and the last core information.

If the core files are generated from an active supervisor module, the number of core files for the service is defined in the service.conf file. There is no upper limit on the total number of core files in the active supervisor module.

To display the core files saved in the system, use the **show cores** command:

Command	Purpose
`switch# show cores`	Displays all the core files.

Verifying First and Last Core Status

You can view specific information about the saved core files. Example 6-16 provides further details on saved core files.

Example 6-16 Regular Service on vdc 2 on Active Supervisor Module

There are five radius core files from vdc2 on the active supervisor module. The second and third oldest files are deleted to comply with the number of core files defined in the service.conf file.

```
switch# show cores vdc vdc2

VDC No Module-num       Process-name      PID     Core-create-time
------ ----------       ------------      ---     ----------------
2      5                radius            6100    Jan 29 01:47
2      5                radius            6101    Jan 29 01:55
2      5                radius            6102    Jan 29 01:55
2      5                radius            6103    Jan 29 01:55
2      5                radius            6104    Jan 29 01:57

switch# show cores vdc vdc2

VDC No Module-num       Process-name      PID     Core-create-time
------ ----------       ------------      ---     ----------------
2      5                radius            6100    Jan 29 01:47
2      5                radius            6103    Jan 29 01:55
2      5                radius            6104    Jan 29 01:57
```

Online System Health Management

The Online Health Management System (OHMS) (system health) is a hardware fault detection and recovery feature. It ensures the general health of switching, services, and supervisor modules in any switch in the Cisco MDS 9000 Family.

This section includes the following topics:

About OHMS

The OHMS monitors system hardware in the following ways:

- The OHMS component running on the active supervisor maintains control over all other OHMS components running on the other modules in the switch.
- The system health application running in the standby supervisor module only monitors the standby supervisor module, if that module is available in the HA standby mode.

The OHMS application launches a daemon process in all modules and runs multiple tests on each module to test individual module components. The tests run at preconfigured intervals, cover all major fault points, and isolate any failing component in the MDS switch. The OHMS running on the active supervisor maintains control over all other OHMS components running on all other modules in the switch.

On detecting a fault, the system health application attempts the following recovery actions:

- Performs additional testing to isolate the faulty component.
- Attempts to reconfigure the component by retrieving its configuration information from persistent storage.
- If unable to recover, sends Call Home notifications, system messages and exception logs; and shuts down and discontinues testing the failed module or component (such as an interface).
- Sends Call Home and system messages and exception logs as soon as it detects a failure.
- Shuts down the failing module or component (such as an interface).
- Isolates failed ports from further testing.
- Reports the failure to the appropriate software component.
- Switches to the standby supervisor module, if an error is detected on the active supervisor module and a standby supervisor module exists in the Cisco MDS switch. After the switchover, the new active supervisor module restarts the active supervisor tests.
- Reloads the switch if a standby supervisor module does not exist in the switch.

- Provides CLI support to view, test, and obtain test run statistics or change the system health test configuration on the switch.
- Performs tests to focus on the problem area.

Each module is configured to run the test relevant to that module. You can change the default parameters of the test in each module as required.

System Health Initiation

By default, the system health feature is enabled in each switch in the Cisco MDS 9000 Family.

To disable or enable this feature in any switch in the Cisco MDS 9000 Family, follow these steps:

	Command	Purpose
Step 1	`switch# config terminal` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# no system health` `System Health is disabled.`	Disables system health from running tests in this switch.
	`switch(config)# system health` `System Health is enabled.`	Enables (default) system health to run tests in this switch.
Step 3	`switch(config)# no system health interface fc8/1` `System health for interface fc8/13 is disabled.`	Disables system health from testing the specified interface.
	`switch(config)# system health interface fc8/1` `System health for interface fc8/13 is enabled.`	Enables (default) system health to test for the specified interface.

Loopback Test Configuration Frequency

Loopback tests are designed to identify hardware errors in the data path in the module(s) and the control path in the supervisors. One loopback frame is sent to each module at a preconfigured frequency—it passes through each configured interface and returns to the supervisor module.

The loopback tests can be run at frequencies ranging from 5 seconds (default) to 255 seconds. If you do not configure the loopback frequency value, the default frequency of 5 seconds is used for all modules in the switch. Loopback test frequencies can be altered for each module.

To configure the frequency of loopback tests for all modules on a switch, follow these steps:

	Command	Purpose
Step 1	`switch# config terminal` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# system health loopback frequency 50` `The new frequency is set at 50 Seconds.`	Configures the loopback frequency to 50 seconds. The default loopback frequency is 5 seconds. The valid range is from 5 to 255 seconds.

Loopback Test Configuration Frame Length

Loopback tests are designed to identify hardware errors in the data path in the module(s) and the control path in the supervisors. One loopback frame is sent to each module at a preconfigured size—it passes through each configured interface and returns to the supervisor module.

The loopback tests can be run with frame sizes ranging from 0 bytes to 128 bytes. If you do not configure the loopback frame length value, the switch generates random frame lengths for all modules in the switch (auto mode). Loopback test frame lengths can be altered for each module.

To configure the frame length for loopback tests for all modules on a switch, follow these steps:

	Command	Purpose
Step 1	`switch# config terminal` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# system health loopback frame-length 128`	Configures the loopback frame length to 128 bytes. The valid range is 0 to 128 bytes.
Step 3	`switch(config)# system health loopback frame-length auto`	Configures the loopback frame length to automatically generate random lengths (default).

To verify the loopback frequency configuration, use the **show system health loopback frame-length** command.

```
switch# show system health loopback frame-length
Loopback frame length is set to auto-size between 0-128 bytes
```

Hardware Failure Action

The failure-action command controls the Cisco NX-OS software from taking any action if a hardware failure is determined while running the tests.

By default, this feature is enabled in all switches in the Cisco MDS 9000 Family—action is taken if a failure is determined and the failed component is isolated from further testing.

Failure action is controlled at individual test levels (per module), at the module level (for all tests), or for the entire switch.

To configure failure action in a switch, follow these steps:

	Command	Purpose
Step 1	`switch# config terminal` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# system health failure-action` `System health global failure action is now enabled.`	Enables the switch to take failure action (default).
Step 3	`switch(config)# no system health failure-action` `System health global failure action now disabled.`	Reverts the switch configuration to prevent failure action being taken.
Step 4	`switch(config)# system health module 1 failure-action` `System health failure action for module 1 is now enabled.`	Enables switch to take failure action for failures in module 1.
Step 5	`switch(config)# no system health module 1 loopback failure-action` `System health failure action for module 1 loopback test is now disabled.`	Prevents the switch from taking action on failures determined by the loopback test in module 1.

Test Run Requirements

Enabling a test does not guarantee that the test will run.

Tests on a specific interface or module only run if you enable system health for all of the following items:

- The entire switch
- The required module
- The required interface

Tip The test will not run if system health is disabled in any combination. If system health is disabled to run tests, the test status shows up as disabled.

Tip If the specific module or interface is enabled to run tests, but is not running the tests due to system health being disabled, then tests show up as enabled (not running).

Tests for a Specified Module

The system health feature in the NX-OS software performs tests in the following areas:

- Active supervisor's in-band connectivity to the fabric.
- Standby supervisor's arbiter availability.
- Bootflash connectivity and accessibility on all modules.
- EOBC connectivity and accessibility on all modules.
- Data path integrity for each interface on all modules.
- Management port's connectivity.
- User-driven test for external connectivity verification, port is shut down during the test (Fibre Channel ports only).
- User-driven test for internal connectivity verification (Fibre Channel and iSCSI ports).

To perform the required test on a specific module, follow these steps:

	Command	Purpose
Step 1	`switch# config terminal` `switch(config)#`	Enters configuration mode.
	Note The following steps can be performed in any order.	
	Note The various options for each test are described in the next step. Each command can be configured in any order. The various options are presented in the same step for documentation purposes.	
Step 2	`switch(config)# system health module 8 bootflash`	Enables the bootflash test on module in slot 8.
	`switch(config)# system health module 8 bootflash frequency 200`	Sets the new frequency of the bootflash test on module 8 to 200 seconds.
Step 3	`switch(config)# system health module 8 eobc`	Enables the EOBC test on module in slot 8.
Step 4	`switch(config)# system health module 8 loopback`	Enables the loopback test on module in slot 8.
Step 5	`switch(config)# system health module 5 management`	Enables the management test on module in slot 5.

Clearing Previous Error Reports

You can clear the error history for Fibre Channel interfaces, iSCSI interfaces, an entire module, or one particular test for an entire module. By clearing the history, you are directing the software to retest all failed components that were previously excluded from tests.

If you previously enabled the failure-action option for a period of time (for example, one week) to prevent OHMS from taking any action when a failure is encountered and after that week you are now ready to start receiving these errors again, then you must clear the system health error status for each test.

Tip The management port test cannot be run on a standby supervisor module.

Use the EXEC-level **system health clear-errors** command at the interface or module level to erase any previous error conditions logged by the system health application. The **bootflash**, the **eobc**, the **inband**, the **loopback**, and the **mgmt** test options can be individually specified for a given module.

The following example clears the error history for the specified Fibre Channel interface:

```
switch# system health clear-errors interface fc 3/1
```

The following example clears the error history for the specified module:

```
switch# system health clear-errors module 3
```

The following example clears the management test error history for the specified module:

```
switch# system health clear-errors module 1 mgmt
```

Performing Internal Loopback Tests

You can run manual loopback tests to identify hardware errors in the data path in the switching or services modules, and the control path in the supervisor modules. Internal loopback tests send and receive FC2 frames to and from the same ports and provide the round-trip time taken in microseconds. These tests are available for Fibre Channel, IPS, and iSCSI interfaces.

Use the EXEC-level **system health internal-loopback** command to explicitly run this test on demand (when requested by the user) within ports for the entire module.

```
switch# system health internal-loopback interface iscsi 8/1
Internal loopback test on interface iscsi8/1 was  successful.
Sent 1 received 1 frames
Round trip time taken is 79 useconds
```

Use the EXEC-level **system health internal-loopback** command to explicitly run this test on demand (when requested by the user) within ports for the entire module and override the frame count configured on the switch.

```
switch# system health internal-loopback interface iscsi 8/1 frame-count 20
Internal loopback test on interface iscsi8/1 was  successful.
Sent 1 received 1 frames
Round trip time taken is 79 useconds
```

Use the EXEC-level **system health internal-loopback** command to explicitly run this test on demand (when requested by the user) within ports for the entire module and override the frame length configured on the switch.

```
switch# system health internal-loopback interface iscsi 8/1 frame-count 32
```

```
Internal loopback test on interface iscsi8/1 was  successful.
Sent 1 received 1 frames
Round trip time taken is 79 useconds
```

Note If the test fails to complete successfully, the software analyzes the failure and prints the following error:

```
External loopback test on interface fc 7/2 failed. Failure reason: Failed to loopback,
analysis complete Failed device ID 3 on module 1
```

Performing External Loopback Tests

You can run manual loopback tests to identify hardware errors in the data path in the switching or services modules, and the control path in the supervisor modules. External loopback tests send and receive FC2 frames to and from the same port or between two ports.

You need to connect a cable (or a plug) to loop the Rx port to the Tx port before running the test. If you are testing to and from the same port, you need a special loop cable. If you are testing to and from different ports, you can use a regular cable. This test is only available for Fibre Channel interfaces.

Use the EXEC-level **system health external-loopback interface** *interface* command to run this test on demand for external devices connected to a switch that is part of a long-haul network.

```
switch# system health external-loopback interface fc 3/1
This will shut the requested interfaces Do you want to continue (y/n)?  [n] y
External loopback test on interface fc3/1 was  successful.
Sent 1 received 1 frames
```

Use the EXEC-level **system health external-loopback source** *interface* **destination interface** *interface* command to run this test on demand between two ports on the switch.

```
switch# system health external-loopback source interface fc 3/1 destination interface fc
3/2
This will shut the requested interfaces Do you want to continue (y/n)?  [n] y
External loopback test on interface fc3/1 and interface fc3/2 was  successful.
Sent 1 received 1 frames
```

Use the EXEC-level **system health external-loopback** *interface* **frame-count** command to run this test on demand for external devices connected to a switch that is part of a long-haul network and override the frame count configured on the switch.

```
switch# system health external-loopback interface fc 3/1 frame-count 10
This will shut the requested interfaces Do you want to continue (y/n)?  [n] y
External loopback test on interface fc3/1 was  successful.
Sent 1 received 1 frames
```

Use the EXEC-level **system health external-loopback** *interface* **frame-length** command to run this test on demand for external devices connected to a switch that is part of a long-haul network and override the frame length configured on the switch.

```
switch# system health external-loopback interface fc 3/1 frame-length 64
This will shut the requested interfaces Do you want to continue (y/n)?  [n] y
External loopback test on interface fc3/1 was  successful.
Sent 1 received 1 frames
```

Use the **system health external-loopback** *interface* **force** command to shut down the required interface directly without a back out confirmation.

```
switch# system health external-loopback interface fc 3/1 force
```

```
External loopback test on interface fc3/1 was  successful.
Sent 1 received 1 frames
```

Note If the test fails to complete successfully, the software analyzes the failure and prints the following error:

```
External loopback test on interface fc 7/2 failed. Failure reason: Failed to loopback,
analysis complete Failed device ID 3 on module 1
```

Performing Serdes Loopbacks

Serializer/Deserializer (serdes) loopback tests the hardware for a port. These tests are available for Fibre Channel interfaces.

Use the EXEC-level **system health serdes-loopback** command to explicitly run this test on demand (when requested by the user) within ports for the entire module.

```
switch# system health serdes-loopback interface fc 3/1
This will shut the requested interfaces Do you want to continue (y/n)?  [n] y
Serdes loopback test passed for module 3 port 1
```

Use the EXEC-level **system health serdes-loopback** command to explicitly run this test on demand (when requested by the user) within ports for the entire module and override the frame count configured on the switch.

```
switch# system health serdes-loopback interface fc 3/1 frame-count 10
This will shut the requested interfaces Do you want to continue (y/n)?  [n] y
Serdes loopback test passed for module 3 port 1
```

Use the EXEC-level **system health serdes-loopback** command to explicitly run this test on demand (when requested by the user) within ports for the entire module and override the frame length configured on the switch.

```
switch# system health serdes-loopback interface fc 3/1 frame-length 32
This will shut the requested interfaces Do you want to continue (y/n)?  [n] y
Serdes loopback test passed for module 3 port 1
```

Note If the test fails to complete successfully, the software analyzes the failure and prints the following error:

```
External loopback test on interface fc 3/1 failed. Failure reason: Failed to loopback,
analysis complete Failed device ID 3 on module 3
```

Interpreting the Current Status

The status of each module or test depends on the current configured state of the OHMS test in that particular module (see Table 6-1).

Table 6-1 OHMS Configured Status for Tests and Modules

Status	Description
Enabled	You have currently enabled the test in this module and the test is not running.
Disabled	You have currently disabled the test in this module.

Table 6-1 OHMS Configured Status for Tests and Modules (continued)

Status	Description
Running	You have enabled the test and the test is currently running in this module.
Failing	This state is displayed if a failure is imminent for the test running in this module—possibility of test recovery exists in this state.
Failed	The test has failed in this module—and the state cannot be recovered.
Stopped	The test has been internally stopped in this module by the Cisco NX-OS software.
Internal failure	The test encountered an internal failure in this module. For example, the system health application is not able to open a socket as part of the test procedure.
Diags failed	The startup diagnostics has failed for this module or interface.
On demand	The system health external-loopback or the system health internal-loopback tests are currently running in this module. Only these two commands can be issued on demand.
Suspended	Only encountered in the MDS 9100 Series due to one oversubscribed port moving to a E or TE port mode. If one oversubscribed port moves to this mode, the other three oversubscribed ports in the group are suspended.

The status of each test in each module is visible when you display any of the **show system health** commands. See the "Displaying System Health" section on page 6-17.

Displaying System Health

Use the **show system health** command to display system-related status information (see Example 6-17 to Example 6-22).

Example 6-17 Displays the Current Health of All Modules in the Switch

```
switch# show system health

Current health information for module 2.

Test                       Frequency    Status          Action
------------------------------------------------------------------
Bootflash                 5 Sec         Running         Enabled
EOBC                      5 Sec         Running         Enabled
Loopback                  5 Sec         Running         Enabled
------------------------------------------------------------------

Current health information for module 6.

Test                       Frequency    Status          Action
------------------------------------------------------------------
InBand                    5 Sec         Running         Enabled
Bootflash                 5 Sec         Running         Enabled
EOBC                      5 Sec         Running         Enabled
Management Port           5 Sec         Running         Enabled
------------------------------------------------------------------
```

Example 6-18 Displays the Current Health of a Specified Module

```
switch# show system health module 8

Current health information for module 8.

Test                       Frequency    Status          Action
------------------------------------------------------------------
Bootflash                  5 Sec        Running         Enabled
EOBC                       5 Sec        Running         Enabled
Loopback                   5 Sec        Running         Enabled
------------------------------------------------------------------
```

Example 6-19 Displays Health Statistics for All Modules

```
switch# show system health statistics

Test statistics for module # 1
-------------------------------------------------------------------------------
Test Name           State            Frequency Run    Pass    Fail CFail Errs
-------------------------------------------------------------------------------
Bootflash           Running             5s    12900   12900       0     0    0
EOBC                Running             5s    12900   12900       0     0    0
Loopback            Running             5s    12900   12900       0     0    0
-------------------------------------------------------------------------------

Test statistics for module # 3
-------------------------------------------------------------------------------
Test Name           State            Frequency Run    Pass    Fail CFail Errs
-------------------------------------------------------------------------------
Bootflash           Running             5s    12890   12890       0     0    0
EOBC                Running             5s    12890   12890       0     0    0
Loopback            Running             5s    12892   12892       0     0    0
-------------------------------------------------------------------------------

Test statistics for module # 5
-------------------------------------------------------------------------------
Test Name           State            Frequency Run    Pass    Fail CFail Errs
-------------------------------------------------------------------------------
InBand              Running             5s    12911   12911       0     0    0
Bootflash           Running             5s    12911   12911       0     0    0
EOBC                Running             5s    12911   12911       0     0    0
Management Port     Running             5s    12911   12911       0     0    0
-------------------------------------------------------------------------------

Test statistics for module # 6
-------------------------------------------------------------------------------
Test Name           State            Frequency Run    Pass    Fail CFail Errs
-------------------------------------------------------------------------------
InBand              Running             5s    12907   12907       0     0    0
Bootflash           Running             5s    12907   12907       0     0    0
EOBC                Running             5s    12907   12907       0     0    0
-------------------------------------------------------------------------------

Test statistics for module # 8
-------------------------------------------------------------------------------
Test Name           State            Frequency Run    Pass    Fail CFail Errs
-------------------------------------------------------------------------------
Bootflash           Running             5s    12895   12895       0     0    0
EOBC                Running             5s    12895   12895       0     0    0
Loopback            Running             5s    12896   12896       0     0    0
-------------------------------------------------------------------------------
```

Example 6-20 Displays Statistics for a Specified Module

```
switch# show system health statistics module 3

Test statistics for module # 3
-------------------------------------------------------------------------------
Test Name           State            Frequency Run    Pass    Fail CFail Errs
-------------------------------------------------------------------------------
Bootflash           Running             5s    12932   12932       0     0    0
EOBC                Running             5s    12932   12932       0     0    0
Loopback            Running             5s    12934   12934       0     0    0
-------------------------------------------------------------------------------
```

Example 6-21 Displays Loopback Test Statistics for the Entire Switch

```
switch# show system health statistics loopback
-----------------------------------------------------------------
Mod Port Status                Run     Pass     Fail   CFail Errs
  1   16 Running             12953    12953        0       0    0
  3   32 Running             12945    12945        0       0    0
  8    8 Running             12949    12949        0       0    0
-----------------------------------------------------------------
```

Example 6-22 Displays Loopback Test Statistics for a Specified Interface

```
switch# show system health statistics loopback interface fc 3/1
-----------------------------------------------------------------
Mod Port Status                Run     Pass     Fail   CFail Errs
  3    1 Running                 0        0        0       0    0
-----------------------------------------------------------------
```

Note Interface-specific counters will remain at zero unless the module-specific loopback test reports errors or failures.

Example 6-23 Displays the Loopback Test Time Log for All Modules

```
switch# show system health statistics loopback timelog
-----------------------------------------------------------------
Mod        Samples     Min(usecs)     Max(usecs)     Ave(usecs)
  1           1872            149            364            222
  3           1862            415            743            549
  8           1865            134            455            349
-----------------------------------------------------------------
```

Example 6-24 Displays the Loopback Test Time Log for a Specified Module

```
switch# show system health statistics loopback module 8 timelog
-----------------------------------------------------------------
Mod        Samples     Min(usecs)     Max(usecs)     Ave(usecs)
  8           1867            134            455            349
-----------------------------------------------------------------
```

On-Board Failure Logging

The Generation 2 Fibre Channel switching modules provide the facility to log failure data to persistent storage, which can be retrieved and displayed for analysis. This on-board failure logging (OBFL) feature stores failure and environmental information in nonvolatile memory on the module. The information will help in post-mortem analysis of failed cards.

This section includes the following topics:

About OBFL

OBFL data is stored in the existing CompactFlash on the module. OBFL uses the persistent logging (PLOG) facility available in the module firmware to store data in the CompactFlash. It also provides the mechanism to retrieve the stored data.

The data stored by the OBFL facility includes the following:

- Time of initial power-on
- Slot number of the card in the chassis
- Initial temperature of the card
- Firmware, BIOS, FPGA, and ASIC versions
- Serial number of the card
- Stack trace for crashes
- CPU hog information
- Memory leak information
- Software error messages
- Hardware exception logs
- Environmental history
- OBFL specific history information
- ASIC interrupt and error statistics history
- ASIC register dumps

Configuring OBFL for the Switch

To configure OBFL for all the modules on the switch, follow these steps:

	Command	Purpose
Step 1	`switch# config terminal` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# hw-module logging onboard`	Enables all OBFL features.
	`switch(config)# hw-module logging onboard cpu-hog`	Enables the OBFL CPU hog events.
	`switch(config)# hw-module logging onboard environmental-history`	Enables the OBFL environmental history.
	`switch(config)# hw-module logging onboard error-stats`	Enables the OBFL error statistics.
	`switch(config)# hw-module logging onboard interrupt-stats`	Enables the OBFL interrupt statistics.
	`switch(config)# hw-module logging onboard mem-leak`	Enables the OBFL memory leak events.
	`switch(config)# hw-module logging onboard miscellaneous-error`	Enables the OBFL miscellaneous information.
	`switch(config)# hw-module logging onboard obfl-log`	Enables the boot uptime, device version, and OBFL history.
	`switch(config)# no hw-module logging onboard`	Disables all OBFL features.

Use the **show logging onboard status** command to display the configuration status of OBFL.

```
switch# show logging onboard status

    Switch OBFL Log:                                        Enabled

    Module:  6 OBFL Log:                                    Enabled
    error-stats                                             Enabled
    exception-log                                           Enabled
    miscellaneous-error                                     Enabled
    obfl-log (boot-uptime/device-version/obfl-history)      Enabled
    system-health                                           Enabled
    stack-trace                                             Enabled
```

Configuring OBFL for a Module

To configure OBFL for specific modules on the switch, follow these steps:

	Command	Purpose
Step 1	`switch# config terminal` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# hw-module logging onboard module 1`	Enables all OBFL features on a module.
	`switch(config)# hw-module logging onboard module 1 cpu-hog`	Enables the OBFL CPU hog events on a module.
	`switch(config)# hw-module logging onboard module 1 environmental-history`	Enables the OBFL environmental history on a module.
	`switch(config)# hw-module logging onboard module 1 error-stats`	Enables the OBFL error statistics on a module.
	`switch(config)# hw-module logging onboard module 1 interrupt-stats`	Enables the OBFL interrupt statistics on a module.
	`switch(config)# hw-module logging onboard module 1 mem-leak`	Enables the OBFL memory leak events on a module.
	`switch(config)# hw-module logging onboard module 1 miscellaneous-error`	Enables the OBFL miscellaneous information on a module.
	`switch(config)# hw-module logging onboard module 1 obfl-log`	Enables the boot uptime, device version, and OBFL history on a module.
	`switch(config)# no hw-module logging onboard module 1`	Disables all OBFL features on a module.

Use the **show logging onboard status** command to display the configuration status of OBFL.

```
switch# show logging onboard status

    Switch OBFL Log:                                        Enabled

    Module:  6 OBFL Log:                                    Enabled
    error-stats                                             Enabled
    exception-log                                           Enabled
    miscellaneous-error                                     Enabled
    obfl-log (boot-uptime/device-version/obfl-history)      Enabled
    system-health                                           Enabled
    stack-trace                                             Enabled
```

Displaying OBFL Logs

To display OBFL information stored in CompactFlash on a module, use the following commands:

Command	Purpose
show logging onboard boot-uptime	Displays the boot and uptime information.
show logging onboard cpu-hog	Displays information for CPU hog events.
show logging onboard device-version	Displays device version information.
show logging onboard endtime	Displays OBFL logs to an end time.
show logging onboard environmental-history	Displays environmental history.
show logging onboard error-stats	Displays error statistics.
show logging onboard exception-log	Displays exception log information.
show logging onboard interrupt-stats	Displays interrupt statistics.
show logging onboard mem-leak	Displays memory leak information.
show logging onboard miscellaneous-error	Displays miscellaneous error information.
show logging onboard module *slot*	Displays OBFL information for a specific module.
show logging onboard obfl-history	Displays history information.
show logging onboard register-log	Displays register log information.
show logging onboard stack-trace	Displays kernel stack trace information.
show logging onboard starttime	Displays OBFL logs from a specified start time.
show logging onboard system-health	Displays system health information.

Clearing the Module Counters

To reset the module couters, follow these steps:

	Command	Purpose
Step 1	`switch# attach module 1` `ModuleX#`	Attaches module 1 to the chasis.
Step 2	`ModuleX# clear asic-cnt all`	Clears the counters for all the devices in the module.
	`ModuleX# clear asic-cnt list-all-devices` `ModuleX# clear asic-cnt device-id` *device-id*	Clears the counters for only the specified device ID. The device ID can vary from 1 through 255.

To reset the counters for all the modules, follow these steps:

	Command	Purpose
Step 1	`switch# debug system internal clear-counters all` `switch#`	Clears the counters for all the modules in the switch.

Note The module counters cannot be cleared using the Device Manager or the Fabric Manager.

This example shows the device IDs of all the devices in a module:

```
switch# attach  module 4
Attaching to module 4 ...
To exit type 'exit', to abort type '$.'
Linux lc04 2.6.10_mvl401-pc_target #1 Tue Dec 16 22:58:32 PST 2008 ppc GNU/Linux
module-4# clear asic-cnt list-all-devices

           Asic Name  |        Device ID
         Stratosphere  |               63
          transceiver  |               46
         Skyline-asic  |               57
           Skyline-ni  |               60
         Skyline-xbar  |               59
          Skyline-fwd  |               58
         Tuscany-asic  |               52
         Tuscany-xbar  |               54
          Tuscany-que  |               55
          Tuscany-fwd  |               53
       Fwd-spi-group   |               73
          Fwd-parser   |               74
                eobc   |               10
           X-Bus IO    |                1
   Power Mngmnt Epld   |               25
```

Default Settings

Table 6-2 lists the default system health and log settings.

Table 6-2 Default System Health and Log Settings

Parameters	Default
Kernel core generation	One module
System health	Enabled
Loopback frequency	5 seconds
Failure action	Enabled

CHAPTER 7

Configuring the Embedded Event Manager

Embedded Event Manager monitors events that occur on your device and takes action to recover or troubleshoot these events, based on your configuration.

This chapter describes how to configure the EEM to detect and handle critical events on a device.

This chapter includes the following sections:

About EEM

This section includes the following topics:

EEM Overview

EEM consists of three major components:

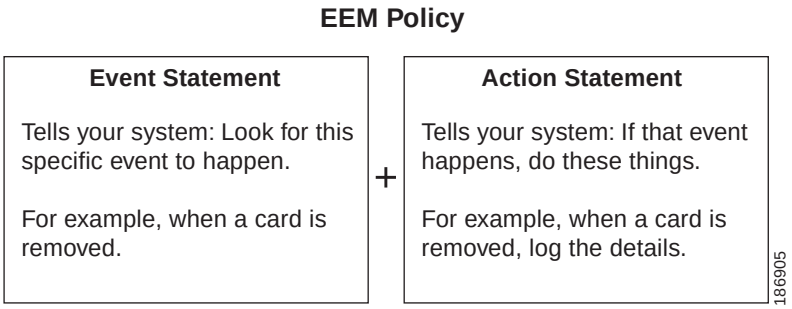
- Event statements—Events to monitor from another Cisco NX-OS component that may require some action, workaround, or notification.
- Action statements —An action that EEM can take, such as sending an e-mail, or disabling an interface, to recover from an event.
- Policies—An event paired with one or more actions to troubleshoot or recover from the event.

Policies

An EEM policy consists of an event statement and one or more action statements. The event statement defines the event to look for as well as the filtering characteristics for the event. The action statement defines the action EEM takes when the event occurs.

Figure 7-1 shows the two basic statements in an EEM policy.

Figure 7-1 EEM Policy Statements



You can configure EEM policies using the CLI or using a VSH script.

Note EEM policy matching is not supported on MDS switches.

EEM maintains event logs on the supervisor.

Cisco NX-OS has a number of preconfigured system policies. These system policies define many common events and actions for the device. System policy names begin with two underscore characters (__).

You can create user policies to suit your network. If you create a user policy, any actions in your policy occur after EEM triggers any system policy actions related to the same event as your policy. To configure a user policy, see the "Defining a User Policy Using the CLI" section on page 7-6.

You can also override some system policies. The overrides that you configure take the place of the system policy. You can override the event or the actions.

Use the **show event manager system-policy** command to view the preconfigured system policies and determine which policies that you can override.

To configure an overriding policy, see the "Overriding a Policy" section on page 7-11.

Note You should use the **show running-config eem** command to check the configuration of each policy. An override policy that consists of an event statement and no action statement triggers no action and no notification of failures.

Note Your override policy should always include an event statement. An override policy without an event statement overrides all possible events in the system policy.

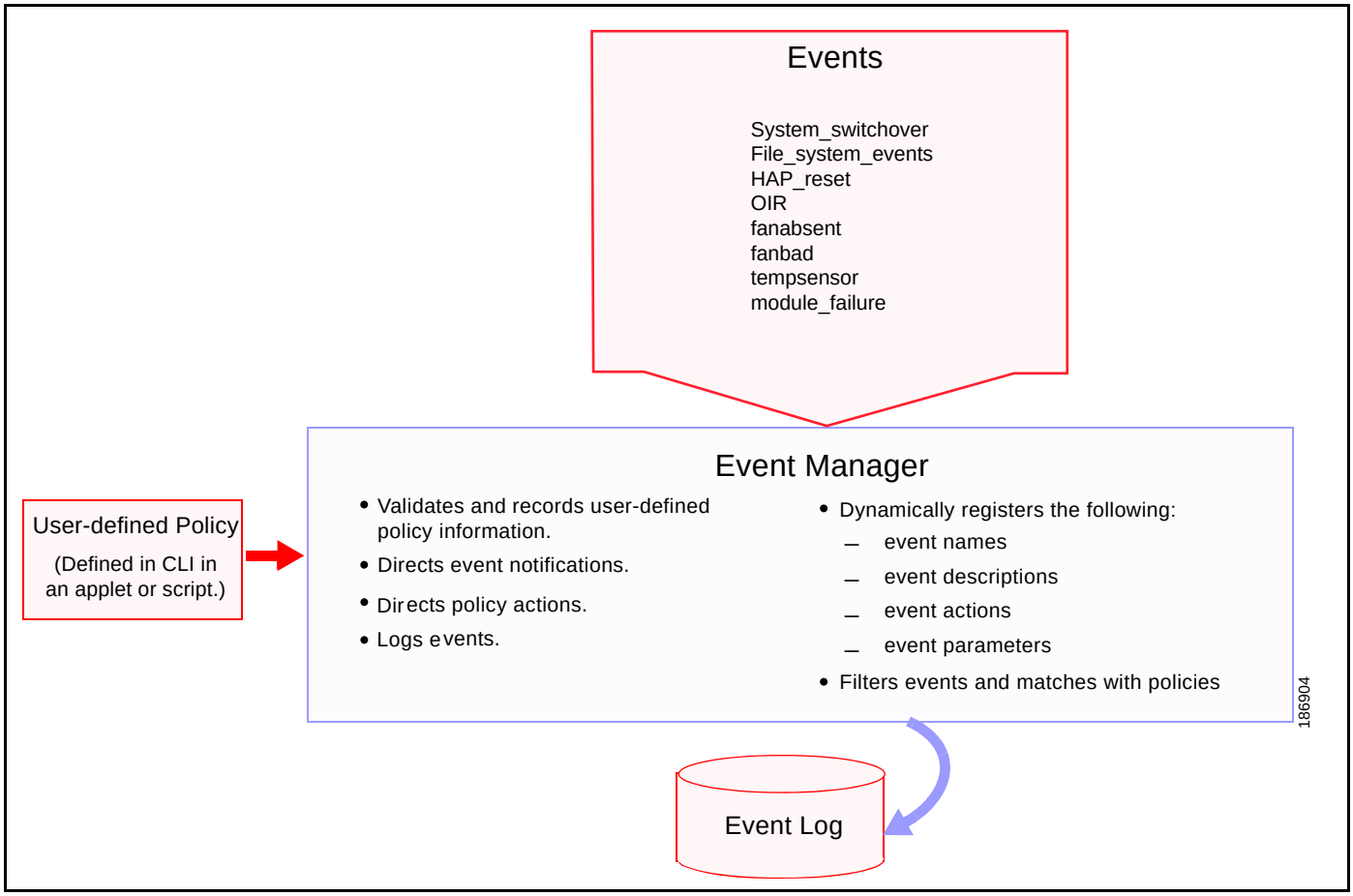
Event Statements

An event is any device activity for which some action, such as a workaround or a notification, should be taken. In many cases, these events are related to faults in the device such as when an interface or a fan malfunctions.

EEM defines event filters so only critical events or multiple occurrences of an event within a specified time period trigger an associated action.

Figure 7-2 shows events that are handled by EEM.

Figure 7-2 EEM Overview



Event statements specify the event that triggers a policy to run. You can configure only one event statement per policy.

EEM schedules and runs policies on the basis of event statements. EEM examines the event and action commands and runs them as defined.

Note If you want to allow the triggered event to process the default actions also, you must explicitly configure an EEM action with “event-default” or “policy-default”, based on the type of policy.

Action Statements

Action statements describe the action triggered by a policy. Each policy can have multiple action statements. If no action is associated with a policy, EEM still observes events but takes no actions.

EEM supports the following actions in action statements:

- Execute any CLI commands.
- Update a counter.
- Log an exception.
- Force the shut down of any module.
- Reload the device.
- Shut down specified modules because the power is over budget.
- Generate a syslog message.
- Generate a Call Home event.
- Generate an SNMP notification.
- Use the default action for the system policy.

Note If you want to allow the triggered event to process the default actions also, you must explicitly configure an EEM action with **event-default** or **policy-default**, based on the type of policy. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM will not allow the CLI command to execute.

Note Verify that your action statements within your user policy or overriding policy do not negate each other or adversely affect the associated system policy.

VSH Script Policies

You can also write policies in a VSH script, using a text editor. These policies have an event statement and action statement(s) just as other policies, and these policies can either augment or override system polices. After you write your script policy, copy it to the device and activate it. To configure a policy in a script, see the “Defining a Policy Using a VSH Script” section on page 7-10.

Environment Variables

You can define environment variables for EEM that are available for all policies. Environment variables are useful for configuring common values that you can use in multiple policies. For example, you can create an environment variable for the IP address of an external e-mail server.

You can use an environment variable in action statements by using the parameter substitution format.

Example 7-1 shows a sample action statement to force a module 1 shutdown, with a reset reason of "EEM action."

Example 7-1 Action Statement

```
switch (config-eem-policy)# action 1.0 forceshut module 1 reset-reason "EEM action"
```

If you define an environment variable for the shutdown reason, called default-reason, you can replace that reset reason with the environment variable, as shown in Example 7-2.

Example 7-2 Action Statement with Environment Variable

```
switch (config-eem-policy)# action 1.0 forceshut module 1 reset-reason $default-reason
```

You can reuse this environment variable in any policy. For more information on environment variables, see the "Defining an Environment Variable" section on page 7-12.

High Availability

Cisco NX-OS supports stateless restarts for EEM. After a reboot or supervisor switchover, Cisco NX-OS applies the running configuration.

Licensing Requirements for EEM

The following table shows the licensing requirements for this feature:

Product	License Requirement
NX-OS	EEM requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you.

Prerequisites for EEM

EEM has the following prerequisites:

- You must have network-admin user privileges to configure EEM.

Configuration Guidelines and Limitations

EEM has the following configuration guidelines and limitations:

- Action statements within your user policy or overriding policy should not negate each other or adversely affect the associated system policy.
- If you want to allow the triggered event to process the default actions also, you must explicitly configure an EEM action with **event-default** or **policy-default**, based on the type of policy. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM will not allow the CLI command to execute.
- An override policy that consists of an event statement and no action statement triggers no action and no notification of failures.
- An override policy without an event statement overrides all possible events in the system policy.

Configuring EEM

This section includes the following topics:

Defining a User Policy Using the CLI

You can define a user policy using the CLI.

This section includes the following topics:

To define a user policy using the CLI, follow these steps:

	Command	Purpose
Step 1	**config t**	Enters configuration mode.
Step 2	**event manager applet** *applet-name*	Registers the applet with EEM and enters applet configuration mode. The *applet-name* can be any case-sensitive alphanumeric string up to 29 characters.
Step 3	**description** *policy-description*	(Optional) Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 4	**event** *event-statement*	Configures the event statement for the policy. See the "Configuring Event Statements" section on page 7-7.
Step 5	**action** *action-statement*	Configures an action statement for the policy. See the "Configuring Action Statements" section on page 7-8. Repeat Step 5 for multiple action statements.

	Command	Purpose
Step 6	**show event manager policy internal** *name*	(Optional) Displays information about the configured policy.
Step 7	**copy running-config startup-config**	(Optional) Saves this configuration change.

Configuring Event Statements

To configure an event statement, use one the following commands in EEM configuration mode:

Command	Purpose
event cli match *expression* [**count** repeats | **time** *seconds*]	Triggers an event if you enter a CLI command that matches the regular expression. The *repeats* range is from 1 to 65000. The time range, in seconds, is from 0 to 4294967295.
event counter name *counter* **entry-val** *entry* **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** |**ne**} [**exit-val** *exit* **exit-op** {**eq** | **ge** | **gt** | **le** | **lt** |**ne**}]	Triggers an event if the counter crosses the entry threshold (based on the entry operation—greater than, less than, and so on.) The event resets immediately. Optionally, you can configure the event to reset after the counter passes the exit threshold. The *counter* name can be any case-sensitive, alphanumeric string up to 28 characters. The *entry* and *exit* value ranges are from 0 to 2147483647.
event fanabsent [**fan** *number*] **time** *seconds*	Triggers an event if a fan is removed from the device for more than the configured time, in seconds. The fan *number* range is dependent on different switches (for example for 9513 switches the range is from1 to 2, for 9506/9509 switches the range is 1). The *seconds* range is from 10 to 64000.
event fanbad [**fan** *number*] **time** *seconds*	Triggers an event if a fan fails for more than the configured time, in seconds. The fan *number* range is dependent on different switches (for example for 9513 switches the range is from1 to 2, for 9506/9509 switches the range is 1). The *seconds* range is from 10 to 64000.
event memory {**critical** | **minor** | **severe**}	Triggers an event if a memory threshold is crossed.
event module-failure type *failure-type* ***module*** {*slot* | **all**} **count** *repeats* [**time** *seconds*]	Triggers an event if a module experiences the failure type configured. The *slot* range is dependent on different switches (for example for 9513 switches the range is from1 to 13, for 9509 switches the range is 1 to 9). The *repeats* range is from 0 to 4294967295. The *seconds* range is from 0 to 4294967295.

Command	Purpose
event oir {**fan** | **module** | **powersupply**} {**anyoir** | **insert** | **remove**} [*number*]	Triggers an event if the configured device element (fan, module, or power supply) is inserted or removed from the device. You can optionally configure a specific fan, module, or power supply number. The *number* range is as follows: • Fan number is dependent on different switches. • Module number is dependent on different switches. • Power supply number range is from 1 to 2.
event policy-default count *repeats* [**time** *seconds*]	Uses the event configured in the system policy. Use this option for overriding policies. The *repeats* range is from 1 to 65000. The *seconds* range is from 0 to 4294967295.
event poweroverbudget	Triggers an event if the power budget exceeds the capacity of the configured power supplies.
event snmp oid *oid* **get-type** {**exact** | **next**} **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** |**ne**} **entry-val** *entry* [**exit-comb** {**and** | **or**}] **exit-op** {**eq** | **ge** | **gt** | **le** | **lt** |**ne**} **exit-val** *exit* **exit-time** *time* **polling-interval** *interval*	Triggers an event if the SNMP OID crosses the entry threshold (based on the entry operation—greater than, less than, and so on.) The event resets immediately, or optionally you can configure the event to reset after the counter passes the exit threshold. The OID is in dotted decimal notation. The *entry* and *exit* value ranges are from 0 to 18446744073709551615. The time range is from 0 to 2147483647. The interval range is from 1 to 2147483647.
event temperature [**module** *slot*] [*sensor number*] **threshold** {**any** | **major** | **minor**}	Triggers an event if the temperature sensor exceeds the configured threshold. The *slot* range is dependent on different switches. The *sensor range* is from 1 to 8 on MDS modules, but current MDS modules use the range from 1 to 3 only, some modules use the range from 1 to 2.

Configuring Action Statements

To configure action statements, use the following commands in EEM configuration mode:

Command	Purpose
action *number*[**.***number2*] **cli** *command1* [command2...] [**local**]	Executes the configured CLI commands. You can optionally execute the commands on the module where the event occurred. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9.
action *number*[**.***number2*] **counter name** *counter* **value** *val* **op** {**dec** | **inc** | **nop** | **set**}	Modifies the counter by the configured value and operation. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The counter name can be any case-sensitive, alphanumeric string up to 28 characters. The *val* can be an integer from 0 to 2147483647 or a substituted parameter.
action *number*[**.***number2*] **event-default**	Executes the default action for the associated event. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9.
action number [.number2] **exceptionlog module** module syserr error **devid** id **errtype** type **errcode** code **phylayer** layer **ports** list **harderror** error [**desc** string]	Logs an exception if the specific conditions are encountered when an EEM applet is triggered.
action *number*[**.***number2*] **forceshut** [**module** *slot* | **xbar** xbar-*number*] **reset-reason** *seconds*	Forces a module, crossbar, or the entire system to shut down. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The *slot* range is dependent on different switches. The *xbar-number* range is from 1 to 2 and is only available on MDS 9513 modules. The reset reason is a quoted alphanumeric string up to 80 characters.
action *number*[**.***number2*] **overbudgetshut** [**module** *slot* [**-** *slot*]]	Forces one or more modules or the entire system to shut down because of a power overbudget issue. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The *slot* range is dependent on different switches.
action *number*[**.***number2*] **policy-default**	Executes the default action for the policy that you are overriding. The action label is in the format number1.number2. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9.

Command	Purpose
action *number*[**.***number2*] **reload** [**module** *slot* [**-** *slot*]]	Forces one or more modules or the entire system to reload. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The *slot* range is dependent on different switches.
action *number*[**.***number2*] **snmp-trap** {[**intdata1** *data* [**intdata2** *data*] [strdata string]}	Sends an SNMP trap with the configured data. *number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The *data* arguments can by any number up to 80 digits. The *string* can be any alphanumeric string up to 80 characters.
action *number*[**.***number2*] **syslog** [**priority** *prio-val*] **msg** *error-message*	Sends a customized syslog message at the configured priority.*number* can be any number up to 16 digits. The range for *number2* is from 0 to 9. The *error-message* can be any quoted alphanumeric string up to 80 characters.

Note If you want to allow the triggered event to process the default actions also, you must explicitly configure an EEM action with **event-default** or **policy-default**, based on the type of policy. For example, if you match a CLI command in a match statement, you must add the **event-default** action statement to the EEM policy or EEM will not allow the CLI command to execute. You can bypass all CLI based EEM policies using **terminal event-manager bypass** command. To revert use **terminal no event-manager bypass** command.

Defining a Policy Using a VSH Script

To define a policy using a VSH script, follow these steps:

Step 1 In a text editor, list the CLI commands that define the policy.

Step 2 Name the text file and save it.

Step 3 Copy the file to the following system directory:

bootflash://eem/user_script_policies

Registering and Activating a VSH Script Policy

To register and activate a policy defined in a VSH script, follow these steps:

	Command	Purpose
Step 1	**config t**	Enters configuration mode.
Step 2	**event manager policy** *policy-script*	Registers and activates an EEM script policy. The *policy-script* can be any case-sensitive alphanumeric string up to 29 characters.
Step 3	**show event manager internal policy** *name*	(Optional) Displays information about the configured policy.
Step 4	**copy running-config startup-config**	(Optional) Saves this configuration change.

Overriding a Policy

To override a system policy, follow these steps:

	Command	Purpose
Step 1	**config t**	Enters configuration mode.
Step 2	**show event manager policy-state** *system-policy*	(Optional) Displays information about the system policy that you want to override, including thresholds. Use the **show event manager system-policy** command to find the system policy names.
Step 3	**event manager applet** *applet-name* **override** *system-policy*	Overrides a system policy and enters applet configuration mode. The *applet-name* can be any case-sensitive alphanumeric string up to 29 characters. The *system-policy* must be one of the existing system policies.
Step 4	**description** *policy-description*	(Optional) Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.
Step 5	**event** *event-statement*	Configures the event statement for the policy. See the "Configuring Event Statements" section on page 7-7.
Step 6	**action** *action-statement*	Configures an action statement for the policy. See the "Configuring Action Statements" section on page 7-8. Repeat Step 6 for multiple action statements.
Step 7	**show event manager policy-state** *name*	(Optional) Displays information about the configured policy.
Step 8	**copy running-config startup-config**	(Optional) Saves this configuration change.

Defining an Environment Variable

To define a variable to serve as a parameter in an EEM policy, follow these steps:

	Command	Purpose
Step 1	**config t**	Enters configuration mode.
Step 2	**event manager environment** *variable-name variable-value*	Creates an environment variable for EEM. The *variable-name* can be any case-sensitive alphanumeric string up to 29 characters. The *variable-value* can be any quoted alphanumeric string up to 39 characters.
Step 3	**show event manager environment**	(Optional) Displays information about the configured environment variables.
Step 4	**copy running-config startup-config**	(Optional) Saves this configuration change.

Verifying EEM Configuration

To display EEM configuration information, perform one of the following tasks:

Command	Purpose
show event manager environment [*variable-name* | **all**]	Displays information about the event manager environment variables.
show event manager event-types [*event* | **all** | **module** *slot*]	Displays information about the event manager event types.
show event manager history events [**detail**] [**maximum** *num-events*] [**severity** {**catastrophic** | **minor** | **moderate** | **severe**}]	Displays the history of events for all policies.
show event manager policy internal [*policy-name*] [**inactive**]	Displays information about the configured policies.
show event manager policy-state *policy-name*	Displays information about policy state, including thresholds.
show event manager script system [*policy-name* | **all**]	Displays information about the script policies.
show event manager system-policy [**all**]	Displays information about the predefined system policies.
show running-config eem	Displays information about the running configuration for EEM.
show startup-config eem	Displays information about the startup configuration for EEM.

EEM Example Configuration

This example overrides the __lcm_module_failure system policy by changing the threshold for just module 3 hitless upgrade failures. This example also sends a syslog message. The settings in the system policy, __lcm_module_failure, apply in all other cases.

```
event manager applet example2 override __lcm_module_failure
 event module-failure type hitless-upgrade-failure module 3 count 2
 action 1 syslog priority errors msg module 3 "upgrade is not a hitless upgrade!"
 action 2 policy-default
```

This example creates an EEM policy that allows the CLI command to execute but triggers an SNMP notification when a user enters configuration mode on the device:

```
event manager applet TEST
event cli match "conf t"
action 1.0 snmp-trap strdata "Confiiguration change"
action 2.0 event-default
```

Note You must add the **event-default** action statement to the EEM policy or EEM will not allow the CLI command to execute.

Default Settings

Table 7-1 lists the default settings for EEM parameters.

Table 7-1 Default EEM Parameters

Parameters	Default
system policies	active

CHAPTER 8

Configuring RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. You can use the RMON alarms and events to monitor Cisco MDS 9000 Family switches running the Cisco SAN-OS Release 2.0(1b) or later or Cisco NX-OS Release 4.1(3) or later software.

This chapter includes the following sections:

About RMON

RMON allows various network agents and console systems to exchange network monitoring data. It is an Internet Engineering Task Force (IETF) standard monitoring specification. You can use the RMON alarms and events to monitor Cisco MDS 9000 Family switches running the Cisco SAN-OS Release 2.0(1b) or later, or Cisco NX-OS 4.1(1) software. RMON is disabled by default, and no events or alarms are configured in the switch.

All switches in the Cisco MDS 9000 Family support the following RMON functions (defined in RFC 2819):

- Alarm—Each alarm monitors a specific management information base (MIB) object for a specified interval. When the MIB object value exceeds a specified value (rising threshold), the alarm condition is set and only one event is triggered regardless of how long the condition exists. When the MIB object value falls below a certain value (falling threshold), the alarm condition is cleared. This allows the alarm to trigger again when the rising threshold is crossed again.
- Event—Determines the action to take when an event is triggered by an alarm. The action can be to generate a log entry, an SNMP trap, or both.

For agent and management information, see the *Cisco MDS 9000 Family MIB Quick Reference.*

For information on an SNMP-compatible network management station, see the *Cisco Fabric Manager System Management Configuration Guide.*

For SNMP security-related CLI configurations, see the "About SNMP Security" section on page 9-1.

Configuring RMON

RMON is disabled by default and no events or alarms are configured in the switch. You can configure your RMON alarms and events by using the CLI or an SNMP-compatible network management station.

Tip We recommend an additional, generic RMON console application on the network management station (NMS) to take advantage of RMON's network management capabilities. Refer to the *Cisco Fabric Manager System Management Configuration Guide*.

Note You must also configure SNMP on the switch to access RMON MIB objects.

RMON Alarm Configuration

Threshold Manager provides a list of common MIB objects to set an RMON threshold and alarm on. The alarm feature monitors a specific MIB object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold).

You can also set an alarm on any MIB object. The specified MIB must be an existing SNMP MIB object in standard dot notation (1.3.6.1.2.1.2.2.1.14.16777216 16 16777216 for ifInOctets.167772161616777216).

Use one of the following options to specify the interval to monitor the MIB variable (ranges from 1 to 4294967295 seconds):

- Use the **delta** option to test the change between samples of a MIB variable.
- Use the **absolute** option to test each MIB variable directly.
- Use the **delta** option to test any MIB objects that are counters.

The range for the **rising threshold** and **falling threshold** values is -2147483647 to 2147483647.

Caution The **falling threshold** must be less than the **rising threshold**.

You can optionally specify the following parameters:

- The event-number to trigger if the rising or falling threshold exceeds the specified limit.
- The owner of the alarm.

To enable RMON alarms, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# rmon alarm 20 1.3.6.1.2.1.2.2.1.14.16777216 2900 delta rising-threshold 15 1 falling-threshold 0 owner test`	Configures RMON alarm number 20 to monitor the 1.3.6.1.2.1.2.2.1.14.16777216 once every 900 seconds until the alarm is disabled and checks the change in the variables rise or fall. If the value shows a MIB counter increase of 15 or more, the software triggers an alarm. The alarm in turn triggers event number 1, which is configured with the RMON event command. Possible events can include a log entry or an SNMP trap. If the MIB value changes by 0, the alarm is reset and can be triggered again.
	`switch(config)# no rmon alarm 2`	Deletes the specified entry from the alarm table.

RMON Event Configuration

To enable RMON events, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# rmon event 2 log trap eventtrap description CriticalErrors owner Test2`	Creates RMON event number 2 to define CriticalErrors and generates a log entry when the event is triggered by the alarm. The user Test2 owns the row that is created in the event table by this command. This example also generates an SNMP trap when the event is triggered.
	`switch(config)# no rmon event 5`	Deletes an entry from the RMON event table.

RMON Verification

Use the **show rmon** and **show snmp** commands to display configured RMON and SNMP information (see Example 8-1 and 8-3).

Example 8-1 Displays Configured RMON Alarms

```
switch# show rmon alarms
Alarm 1 is active, owned by admin
 Monitors 1.3.6.1.2.1.2.2.1.16.16777216 every 1 second(s)
 Taking delta samples, last value was 0
 Rising threshold is 1, assigned to event 0
 Falling threshold is 0, assigned to event 0
 On startup enable rising or falling alarm
```

Example 8-2 Displays Configured RMON High Capacity Alarms

```
switch# show rmon hcalarms
High Capacity Alarm 10 is active, owned by Testuser
 Monitors 1.3.6.1.2.1.31.1.1.1.6.16785408 every 300 second(s)
 Taking absolute samples, last value was 0 (valuePositive)
 Rising threshold low is 4294967295 & high is 15 (valuePositive)
 Rising threshold assigned to event 1
 Falling threshold low is 0 & high is 0 (valueNotAvailable)
 Falling threshold assigned to event 0
 On startup enable rising alarm
 Number of Failed Attempts is 0
```

Note High capacity RMON alarms can be configured using the CISCO-HC-ALARM-MIB. See the *Cisco MDS 9000 Family MIB Quick Reference*.

Example 8-3 Displays Configured RMON Events

```
switch# show rmon events
Event 2 is active, owned by Test2
 Description is CriticalErrors
 Event firing causes log and trap to community eventtrap, last fired 0
Event 500 is active, owned by admin
 Description is
 Event firing causes log, last fired 138807208
```

Default Settings

Table 8-1 lists the default settings for all RMON features in any switch.

Table 8-1 Default RMON Settings

Parameters	Default
RMON alarms	Disabled
RMON events	Disabled

CHAPTER 9

Configuring SNMP

The CLI and SNMP use common roles in all switches in the Cisco MDS 9000 Family. You can use SNMP to modify a role that was created using the CLI and vice versa.

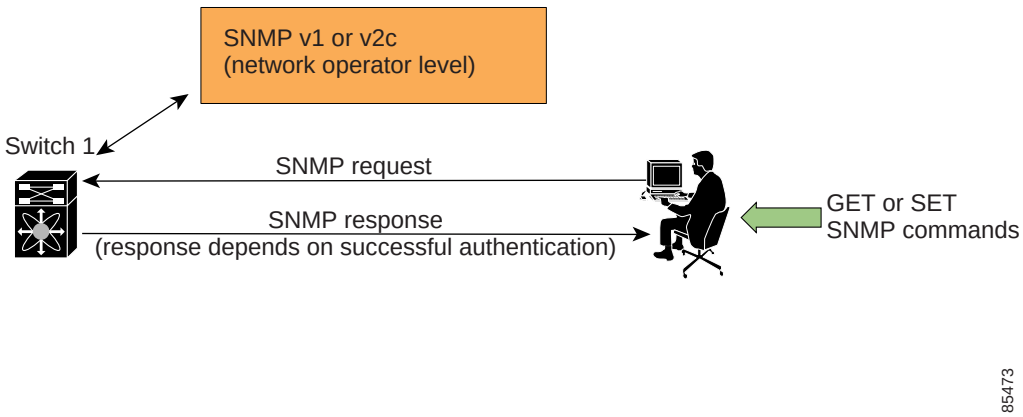
Users, passwords, and roles for all CLI and SNMP users are the same. A user configured through the CLI can access the switch using SNMP (for example, the Fabric Manager or the Device Manager) and vice versa.

This chapter includes the following sections:

About SNMP Security

SNMP is an application layer protocol that facilitates the exchange of management information between network devices. In all Cisco MDS 9000 Family switches, three SNMP versions are available: SNMPv1, SNMPv2c, and SNMPv3 (see Figure 9-1).

Figure 9-1 SNMP Security



This section includes the following topics:

SNMP Version 1 and Version 2c

SNMP Version 1 (SNMPv1) and SNMP Version 2c (SNMPv2c) use a community string match for user authentication. Community strings provided a weak form of access control in earlier versions of SNMP. SNMPv3 provides much improved access control using strong authentication and should be preferred over SNMPv1 and SNMPv2c wherever it is supported.

SNMP Version 3

SNMP Version 3 (SNMPv3) is an interoperable standards-based protocol for network management. SNMPv3 provides secure access to devices by a combination of authenticating and encrypting frames over the network. The security features provided in SNMPv3 are:

- Message integrity—Ensures that a packet has not been tampered with in-transit.
- Authentication—Determines the message is from a valid source.
- Encryption—Scrambles the packet contents to prevent it from being seen by unauthorized sources.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Assigning SNMP Switch Contact and Location Information

You can assign the switch contact information, which is limited to 32 characters (without spaces) and the switch location.

To configure contact and location information, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# snmp-server contact NewUser`	Assigns the contact name for the switch.
	`switch(config)# no snmp-server contact NewUser`	Deletes the contact name for the switch.
Step 3	`switch(config)# snmp-server location SanJose`	Assigns the switch location.
	`switch(config)# no snmp-server location SanJose`	Deletes the switch location.

SNMPv3 CLI User Management and AAA Integration

The Cisco NX-OS software implements RFC 3414 and RFC 3415, including user-based security model (USM) and role-based access control. While SNMP and the CLI have common role management and share the same credentials and access privileges, the local user database was not synchronized in earlier releases.

SNMPv3 user management can be centralized at the AAA server level. This centralized user management allows the SNMP agent running on the Cisco MDS switch to leverage the user authentication service of the AAA server. Once user authentication is verified, the SNMP PDUs are processed further. The AAA server also is used to store user group names. SNMP uses the group names to apply the access/role policy that is locally available in the switch.

This section includes the following topics:

CLI and SNMP User Synchronization

Any configuration changes made to the user group, role, or password results in database synchronization for both SNMP and AAA.

To create an SNMP or CLI user, use either the **username** or **snmp-server user** commands.

- The auth passphrase specified in the **snmp-server user** command is synchronized as the password for the CLI user.
- The password specified in the **username** command is synchronized as the auth and priv passphrases for the SNMP user.

Users are synchronized as follows:

- Deleting a user using either command results in the user being deleted for both SNMP and the CLI.
- User-role mapping changes are synchronized in SNMP and the CLI.

Note When the passphrase/password is specified in localized key/encrypted format, the password is not synchronized.

- Existing SNMP users continue to retain the auth and priv passphrases without any changes.
- If the management station creates an SNMP user in the usmUserTable, the corresponding CLI user is created without any password (login is disabled) and will have the network-operator role.

Restricting Switch Access

You can restrict access to a Cisco MDS 9000 Family switch using IP Access Control Lists (IP-ACLs).

Group-Based SNMP Access

Note Because *group* is a standard SNMP term used industry-wide, we refer to role(s) as group(s) in this SNMP section.

SNMP access rights are organized by groups. Each group in SNMP is similar to a role through the CLI. Each group is defined with three accesses: read access, write access, and notification access. Each access can be enabled or disabled within each group.

You can begin communicating with the agent once your user name is created, your roles are set up by your administrator, and you are added to the roles.

Creating and Modifying Users

You can create users or modify existing users using SNMP, Fabric Manager, or the CLI.

- SNMP—Create a user as a clone of an existing user in the usmUserTable on the switch. Once you have created the user, change the cloned secret key before activating the user. Refer to RFC 2574.
- Fabric Manager.
- CLI—Create a user or modify an existing user using the **snmp-server user** command.

A network-operator and network-admin roles are available in a Cisco MDS 9000 Family switch. There is also a default-role if you want to use the GUI (Fabric Manager and Device Manager). You can also use any role that is configured in the Common Roles database.

Tip All updates to the CLI security database and the SNMP user database are synchronized. You can use the SNMP password to log into either Fabric Manager or Device Manager. However, after you use the CLI password to log into Fabric Manager or Device Manager, you must use the CLI password for all future logins. If a user exists in both the SNMP database and the CLI database before upgrading to Cisco MDS SAN-OS Release 2.0(1b), then the set of roles assigned to the user becomes the union of both sets of roles after the upgrade.

This section includes the following topics:

About AES Encryption-Based Privacy

The Advanced Encryption Standard (AES) is the symmetric cipher algorithm. The Cisco NX-OS software uses AES as one of the privacy protocols for SNMP message encryption and conforms with RFC 3826.

The **priv** option offers a choice of DES or 128-bit AES encryption for SNMP security encryption. The **priv** option along with the **aes-128** token indicates that this privacy password is for generating a 128-bit AES key. The AES priv password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 characters. If you use the localized key, you can specify a maximum of 130 characters.

Note For an SNMPv3 operation using the external AAA server, user configurations in the external AAA server require AES to be the privacy protocol to use SNMP PDU encryption.

Configuring SNMP Users from the CLI

The passphrase specified in the **snmp-server user** command and the **username** command are synchronized.

To create or modify SNMP users from the CLI, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server user joe network-admin auth sha abcd1234`	Creates or modifies the settings for a user (joe) in the network-admin role using the HMAC-SHA-96 authentication password (abcd1234).
	`switch(config)# snmp-server user sam network-admin auth md5 abcdefgh`	Creates or modifies the settings for a user (sam) in the network-admin role using the HMAC-MD5-96 authentication password (abcdefgh).
	`switch(config)# snmp-server user Bill network-admin auth sha abcd1234 priv abcdefgh`	Creates or modifies the settings for a user (Bill) in the network-admin role using the HMAC-SHA-96 authentication level and privacy encryption parameters.
	`switch(config)# no snmp-server user usernameA`	Deletes the user (usernameA) and all associated parameters.
	`switch(config)# no snmp-server usam role vsan-admin`	Deletes the specified user (usam) from the vsan-admin role.
	`switch(config)# snmp-server user user1 network-admin auth md5 0xab0211gh priv 0x45abf342 localizedkey`	Specifies the password to be in localized key format (RFC 2574). The localized key is provided in hexadecimal format (for example, 0xacbdef).
	`switch(config)# snmp-server  user user2 auth md5 asdgfsadf priv aes-128 asgfsgkhkj`	Configures the user2 with the MD5 authentication protocol and AES-128 privacy protocol.
Step 3	`switch(config)# snmp-server user joe sangroup`	Adds the specified user (joe) to the sangroup role.
	`switch(config)# snmp-server user joe techdocs`	Adds the specified user (joe) to the techdocs role.

To create or modify passwords for SNMP users from the CLI, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server user user1 role1 auth md5 0xab0211gh priv 0x45abf342 localizedkey`	Specifies the password to be in localized key format using the DES option for security encryption.
	`switch(config)# snmp-server user user1 role2 auth sha 0xab0211gh priv aes-128 0x45abf342 localizedkey`	Specifies the password to be in localized key format using the 128-bit AES option for security encryption

Caution Avoid using the **localizedkey** option when configuring an SNMP user from the CLI. The localized keys are not portable across devices as they contain device engine ID information. If a configuration file is copied to the device, the passwords may not be set correctly if the configuration file was generated at a different device. Explicitly configure the desired passwords after copying the configuration into the device. Passwords specified with the **localizedkey** option are limited to 130 characters.

Note The **snmp-server user** command takes the engineID as an additional parameter. The engineID creates the notification target user (see the "Configuring the Notification Target User" section on page 9-13). If the engineID is not specified, the local user is created.

Enforcing SNMPv3 Message Encryption

By default the SNMP agent allows the securityLevel parameters of authNoPriv and authPriv for the SNMPv3 messages that use user-configured SNMPv3 message encryption with auth and priv keys.

To enforce the message encryption for a user, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server user testUser enforcePriv`	Enforces the message encryption for SNMPv3 messages using this user. **Note** You can only use this command for previously existing users configured with both auth and priv keys. When the user is configured to enforce privacy, for any SNMPv3 PDU request using securityLevel parameter of either noAuthNoPriv or authNoPriv, the SNMP agent responds with authorizationError.
	`switch(config)# no snmp-server user testUser enforcePriv`	Disables SNMPv3 message encryption enforcement.

Alternatively, you can enforce the SNMPv3 message encryption globally on all the users using the following commands:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server globalEnforcePriv`	Enforces the SNMPv3 message encryption for all the users on the switch.
	`switch(config)# no snmp-server globalEnforcePriv`	Disables global SNMPv3 message encryption enforcement.

Assigning SNMPv3 Users to Multiple Roles

The SNMP server user configuration is enhanced to accommodate multiple roles (groups) for SNMPv3 users. After the initial SNMPv3 user creation, you can map additional roles for the user.

Note Only users belonging to a network-admin role can assign roles to other users.

To configure multiple roles for SNMPv3 users from the CLI, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server user NewUser role1`	Creates or modifies the settings for an SNMPv3 user (NewUser) for the role1 role.
	`switch(config)# snmp-server user NewUser role2`	Creates or modifies the settings for an SNMPv3 user (NewUser) for the role2 role.
	`switch(config)# no snmp-server user User5 role2`	Removes role2 for the specified user (User5).

Adding or Deleting Communities

You can configure read-only or read-write access for SNMPv1 and SNMPv2 users. Refer to RFC 2576.

To create an SNMPv1 or SNMPv2c community, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# snmp-server community snmp_Community ro`	Adds read-only access for the specified SNMP community.
	`switch(config)# snmp-server community snmp_Community rw`	Adds read-write access for the specified SNMP community.
	`switch(config)# no snmp-server community snmp_Community`	Deletes access for the specified SNMP community (default).

SNMP Trap and Inform Notifications

You can configure the Cisco MDS switch to send notifications to SNMP managers when particular events occur.

Note Use the SNMP-TARGET-MIB to obtain more information on the destinations to which notifications are to be sent either as traps or as informs. Refer to the *Cisco MDS 9000 Family MIB Quick Reference*.

This section includes the following topics:

Tip The SNMPv1 option is not available with the **snmp-server host** *ip-address* **informs** command.

Configuring SNMPv2c Notifications

To configure SNMPv2c notifications using IPv4, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server host 171.71.187.101 traps version 2c private udp-port 1163`	Configures the specified host to receive SNMPv2c traps using SNMPv2c community string (private).
	`switch(config)# no snmp-server host 171.71.187.101 traps version 2c private udp-port 2162`	Prevents the specified host from receiving SNMPv2c traps on the configured UDP port using SNMPv2c community string (private).
Step 3	`switch(config)# snmp-server host 171.71.187.101 informs version 2c private udp-port 1163`	Configures the specified host to receive SNMPv2c informs using SNMPv2c community string (private).
	`switch(config)# no snmp-server host 171.71.187.101 informs version 2c private udp-port 2162`	Prevents the specified host from receiving SNMPv2c informs on the configured UDP port using SNMPv2c community string (private).

To configure SNMPv2c notifications using IPv6, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server host 2001:0DB8:800:200C::417A traps version 2c private udp-port 1163`	Configures the specified host to receive SNMPv2c traps using SNMPv2c community string (private).
	`switch(config)# no snmp-server host 2001:0DB8:800:200C::417A traps version 2c private udp-port 2162`	Prevents the specified host from receiving SNMPv2c traps on the configured UDP port using SNMPv2c community string (private).
Step 3	`switch(config)# snmp-server host 2001:0DB8:800:200C::417A informs version 2c private udp-port 1163`	Configures the specified host to receive SNMPv2c informs using SNMPv2c community string (private).
	`switch(config)# no snmp-server host 2001:0DB8:800:200C::417A informs version 2c private udp-port 2162`	Prevents the specified host from receiving SNMPv2c informs on the configured UDP port using SNMPv2c community string (private).

Note Switches can forward events (SNMP traps and informs) up to 10 destinations.

Configuring SNMPv3 Notifications

To configure SNMPv3 notifications using IPv4, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server host 16.20.11.14 traps version 3 noauth testuser udp-port 1163`	Configures the specified host to receive SNMPv3 traps using SNMPv3 user (testuser) and securityLevel of noAuthNoPriv.
	`switch(config)# snmp-server host 16.20.11.14 informs version 3 auth testuser udp-port 1163`	Configures the specified host to receive SNMPv3 informs using SNMPv3 user (testuser) and securityLevel of AuthNoPriv.
	`switch(config)# snmp-server host 16.20.11.14 informs version 3 priv testuser udp-port 1163`	Configures the specified host to receive SNMPv3 informs using SNMPv3 user (testuser) and securityLevel of AuthPriv.
	`switch(config)# no snmp-server host 172.18.2.247 informs version 3 testuser noauth udp-port 2162`	Prevents the specified host from receiving SNMPv3 informs.

To configure SNMPv3 notifications using IPv6, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server host 2001:0DB8:800:200C::417A traps version 3 noauth testuser udp-port 1163`	Configures the specified host to receive SNMPv3 traps using SNMPv3 user (testuser) and securityLevel of noAuthNoPriv.
	`switch(config)# snmp-server host 2001:0DB8:800:200C::417A informs version 3 auth testuser udp-port 1163`	Configures the specified host to receive SNMPv3 informs using SNMPv3 user (testuser) and securityLevel of AuthNoPriv.
	`switch(config)# snmp-server host 2001:0DB8:800:200C::417A informs version 3 priv testuser udp-port 1163`	Configures the specified host to receive SNMPv3 informs using SNMPv3 user (testuser) and securityLevel of AuthPriv.
	`switch(config)# no snmp-server host 2001:0DB8:800:200C::417A informs version 3 testuser noauth udp-port 2162`	Prevents the specified host from receiving SNMPv3 informs.

Enabling SNMP Notifications

Notifications (traps and informs) are system alerts that the switch generates when certain events occur. You can enable or disable notifications. By default, no notification is defined or issued. If a notification name is not specified, all notifications are disabled or enabled.

With the SNMP central infra feature, you can add the traps that need to be enabled or disabled. The MIB CISCO-NOTIFICATION-CONTROL-MIB is supported to enable the use of a MIB browser to control notification generation.

You can enable or disable the supported traps at the following levels:

- Switch level—You can use **snmp-server enable traps** command to enable all the traps in the supported MIBs at the switch level.
- Feature level—You can use **snmp-server enable traps** command with the feature name to enable traps at the feature level.

```
switch =>snmp-server enable traps callhome ?
event-notify    Callhome External Event Notification
smtp-send-fail  SMTP Message Send Fail notification
```

- Individual traps - You can use **snmp-server enable traps** command with the feature name to enable traps at the individual level.

```
switch =>snmp-server enable traps callhome event-notify ?
```

Table 9-1 lists the CLI commands that enable the notifications for Cisco NX-OS MIBs.

Note The **snmp-server enable traps** CLI command enables both traps and informs, depending on how you configured SNMP. See the notifications displayed with the **snmp-server host** CLI command.

Table 9-1 Enabling SNMP Notifications

MIB	Fabric Manager Check boxes
CISCO-ENTITY-FRU-CONTROL-MIB	Select the **Other** tab and check **FRU Changes**.
CISCO-FCC-MIB	Select the **Other** tab and check **FCC**.
CISCO-DM-MIB	Select the **FC** tab and check **Domain Mgr RCF**.
CISCO-NS-MIB	Select the **FC** tab and check **Name Server**.
CISCO-FCS-MIB	Select the **Other** tab and check **FCS Rejects**.
CISCO-FDMI-MIB	Select the **Other** tab and check **FDMI**.
CISCO-FSPF-MIB	Select the **FC** tab and check **FSPF Neighbor Change**.
CISCO-LICENSE-MGR-MIB	Select the **Other** tab and check **License Manager**.
CISCO-IPSEC-SIGNALLING-MIB	Select the **Other** tab and check **IPSEC**.
CISCO-PSM-MIB	Select the **Other** tab and check **Port Security**.
CISCO-RSCN-MIB	Select the **FC** tab and check **RSCN ILS**, and **RCSN ELS**.
SNMPv2-MIB	Select the **Other** tab and check **SNMP AuthFailure**.
VRRP-MIB, CISCO-IETF-VRRP-MIB	Select the **Other** tab and check **VRRP.**
CISCO-ZS-MIB	Select the **FC** tab and check **Zone Rejects**, **Zone Merge Failures**, **Zone Merge Successes**, **Zone Default Policy Change**, and **Zone Unsuppd Mode**.

The following notifications are enabled by default:

- entity fru
- license
- link ietf-extended

All other notifications are disabled by default.

To enable individual notifications, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server enable traps fcdomain`	Enables the specified SNMP (fcdomain) notification.
	`switch(config)# no snmp-server enable traps`	Disables the specified SNMP notification. If a notification name is not specified, all notifications are disabled.

You can use the **show snmp trap** command to display all the notifications and their status.

```
switch# show snmp trap
--------------------------------------------------------------------------------
Trap type                                                              Enabled
--------------------------------------------------------------------------------
entity               : entity_mib_change                               Yes
entity               : entity_module_status_change                     Yes
```

```
entity              : entity_power_status_change                Yes
entity              : entity_module_inserted                    Yes
entity              : entity_module_removed                     Yes
entity              : entity_unrecognised_module                Yes
entity              : entity_fan_status_change                  Yes
entity              : entity_power_out_change                   Yes
link                : linkDown                                  Yes
link                : linkUp                                    Yes
link                : extended-linkDown                         Yes
link                : extended-linkUp                           Yes
link                : cieLinkDown                               Yes
link                : cieLinkUp                                 Yes
link                : connUnitPortStatusChange                  Yes
link                : fcTrunkIfUpNotify                         Yes
link                : fcTrunkIfDownNotify                       Yes
link                : delayed-link-state-change                 Yes
link                : fcot-inserted                             Yes
link                : fcot-removed                              Yes
callhome            : event-notify                              No
callhome            : smtp-send-fail                            No
cfs                 : state-change-notif                        No
cfs                 : merge-failure                             No
fcdomain            : dmNewPrincipalSwitchNotify                No
fcdomain            : dmDomainIdNotAssignedNotify               No
fcdomain            : dmFabricChangeNotify                      No
rf                  : redundancy_framework                      Yes
aaa                 : server-state-change                       No
license             : notify-license-expiry                     Yes
license             : notify-no-license-for-feature             Yes
license             : notify-licensefile-missing                Yes
license             : notify-license-expiry-warning             Yes
scsi                : scsi-disc-complete                        No
fcns                : reject-reg-req                            No
fcns                : local-entry-change                        No
fcns                : db-full                                   No
fcns                : remote-entry-change                       No
rscn                : rscnElsRejectReqNotify                    No
rscn                : rscnIlsRejectReqNotify                    No
rscn                : rscnElsRxRejectReqNotify                  No
rscn                : rscnIlsRxRejectReqNotify                  No
fcs                 : request-reject                            No
fcs                 : discovery-complete                        No
fctrace             : route                                     No
zone                : request-reject1                           No
zone                : merge-success                             No
zone                : merge-failure                             No
zone                : default-zone-behavior-change              No
zone                : unsupp-mem                                No
port-security       : fport-violation                           No
port-security       : eport-violation                           No
port-security       : fabric-binding-violation                  No
vni                 : virtual-interface-created                 No
vni                 : virtual-interface-removed                 No
vsan                : vsanStatusChange                          No
vsan                : vsanPortMembershipChange                  No
fspf                : fspfNbrStateChangeNotify                  No
upgrade             : UpgradeOpNotifyOnCompletion               No
upgrade             : UpgradeJobStatusNotify                    No
feature-control     : FeatureOpStatusChange                     No
vrrp                : cVrrpNotificationNewMaster                No
fdmi                : cfdmiRejectRegNotify                      No
snmp                : authentication                            No
```

Configuring the Notification Target User

You must configure a notification target user on the switch for sending SNMPv3 inform notifications to the SNMP manager.

To configure the notification target user, use the following command:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# snmp-server user testusr auth md5 xyub20gh priv xyub20gh engineID 00:00:00:63:00:01:00:a1:ac:15:10:03`	Configures the notification target user with the specified credentials for the SNMP manager with the specified engine ID.
	`switch(config)# no snmp-server user testusr auth md5 xyub20gh priv xyub20gh engineID 00:00:00:63:00:01:00:a1:ac:15:10:03`	Removes the notification target user.

The credentials of the notification target user are used for encrypting the SNMPv3 inform notification messages to the configured SNMPmanager (as in the **snmp-server host** command).

Note For authenticating and decrypting the received INFORM PDU, the SNMP manager should have the same user credentials in its local configuration data store of users.

Configuring LinkUp/LinkDown Notifications for Switches

You can configure which linkUp/linkDown notifications to enable on switches. You can enable the following types of linkUp/linkDown notifications:

- Cisco—Only notifications (cieLinkUp, cieLinkDown) defined in CISCO-IF-EXTENSION-MIB.my are sent for an interface, if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface.
- IETF—Only notifications (linkUp, linkDown) defined in IF-MIB are sent for an interface, if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface. Only the varbinds defined in the notification definition are sent with the notifications.
- IEFT extended—Only notifications (linkUp, linkDown) defined in IF-MIB are sent for an interface, if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface. In addition to the varbinds defined in the notification definition, varbinds defined in the IF-MIB specific to the Cisco Systems implementation are sent. This is the default setting.
- IEFT Cisco—Only notifications (linkUp, linkDown) defined in IF-MIB and notifications (cieLinkUp, cieLinkDown) defined in CISCO-IF-EXTENSION-MIB.my are sent for an interface, if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface. Only the varbinds defined in the notification definition are sent with the linkUp and linkDown notifications.
- IEFT extended Cisco—Only notifications (linkUp, linkDown) defined in IF-MIB and notifications (cieLinkUp, cieLinkDown) defined in CISCO-IF-EXTENSION-MIB.my are sent for an interface, if ifLinkUpDownTrapEnable (defined in IF-MIB) is enabled for that interface. In addition to the varbinds defined in linkUp and linkDown notification definition, varbinds defined in the IF-MIB specific to the Cisco Systems implementation are sent with the linkUp and linkDown notifications.

Note For more information on the varbinds defined in the IF-MIB specific to the Cisco Systems implementation, refer to the *Cisco MDS 9000 Family MIB Quick Reference.*

To configure the linkUp/linkDown notification for a switch using NX-OS Release 4.1(x) and earlier, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 1	`switch(config)# snmp-server enable traps link`	Enables (default) only IETF extended linkUp/linkDown notifications.
	`switch(config)# snmp-server enable traps link cisco`	Enables only Cisco Systems defined cieLinkUp/cieLinkDown notifications.
	`switch(config)# snmp-server enable traps link ietf`	Enables only IETF linkUp/linkDown notifications.
	`switch(config)# snmp-server enable traps link ietf-extended`	Enables (default) only IETF extended linkUp/linkDown notifications with extra varbinds.
	`switch(config)# snmp-server enable traps link ietf cisco`	Enables IETF (linkUp/linkDown) and Cisco Systems defined (cieLinkUp/cieLinkDown) notifications.
	`switch(config)# snmp-server enable traps link ietf-extended cisco`	Enables IEFT (linkUp/linkDown) notifications with extra varbinds and Cisco Systems defined (cieLinkUp/cieLinkDown) notifications.
	`switch(config)# no snmp-server enable traps link`	Reverts to the default setting (IETF extended).

Note If both IETF and IETF extended are enabled, the **show snmp traps** command displays both as enabled. However, as a trap, you will receive only one trap with IETF extended payload.

To configure the linkUp/linkDown notification for a switch using NX-OS Release 4.2(1) and later, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 1	`switch(config)# snmp-server enable traps link`	Enables (default) only IETF extended linkUp/linkDown notifications.
	`switch(config)# snmp-server enable traps link cieLinkDown`	Enables Cisco extended link state down notification.
	`switch(config)# snmp-server enable traps link cieLinkUp`	Enables Cisco extended link state up notification.
	`switch(config)# snmp-server enable traps link connUnitPortStatusChange`	Enables FCMGMT The overall status of the connectivity unit Notification.
	`switch(config)# snmp-server enable traps link delayed-link-state-change`	Enables Delayed link state change.
	`switch(config)# snmp-server enable traps link extended-linkDown`	Enables IETF extended link state down notification.
	`switch(config)# snmp-server enable traps link extended-linkUp`	Enables IETF extended link state down notification.
	`switch(config)# snmp-server enable traps link fcTrunkIfDownNotify`	Enables FCFE Link state down notification.
	`switch(config)# snmp-server enable traps link fcTrunkIfUpNotify`	Enables FCFE Link state up notification.
	`switch(config)# snmp-server enable traps link fcot-inserted`	Enables FCOT info trap.
	`switch(config)# snmp-server enable traps link fcot-removed`	Enables FCOT info trap.
	`switch(config)# snmp-server enable traps link linkDown`	Enables IETF Link state down notification.
	`switch(config)# snmp-server enable traps link linkUp`	Enables IETF Link state up notification.
	`switch(config)# no snmp-server enable traps link`	Reverts to the default setting (IETF extended).

Configuring Up/Down SNMP Link-State Traps for Interfaces

By default, SNMP link-state traps are enabled for all interfaces. Whenever a link toggles its state from Up to Down or vice versa, an SNMP trap is generated.

In some instances, you may find that you have numerous switches with hundreds of interfaces, many of which do not require monitoring of the link state. In such cases, you may elect to disable link-state traps.

To disable SNMP link-state traps for specific interfaces, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.

Command	Purpose
Step 2 `switch(config)# interface bay 6`	Specifies the interface on which to disable SNMP link-state traps.
`switch(config-if)# no link-state-trap`	Disables SNMP link-state traps for the interface.
`switch(config-if)# link-state-trap`	Enables SNMP link-state traps for the interface.

Whenever you disable an SNMP link-state trap for an interface, the command is also added to the running configuration of the system. To view the running configuration, enter the **show running-config** command for the interface.

```
switch# show running-config
version 3.1(2)
....
interface bay5
interface bay6
  no link-state-trap <-----command is added to the running configuration for the interface
interface bay7...
```

To view the SNMP link-state trap configuration for a particular interface, enter the **show interface** command.

```
switch# show interface bay 6
bay6 is down (Administratively down)
    Hardware is Fibre Channel
    Port WWN is 20:0b:00:05:30:01:70:2c
    Admin port mode is auto, trunk mode is on
    snmp link-state traps are disabled

    Port vsan is 1
    Receive data field Size is 2112
    Beacon is turned off
    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 frames input, 0 bytes
        0 discards, 0 errors
        0 CRC,  0 unknown class
        0 too long, 0 too short
      0 frames output, 0 bytes
        0 discards, 0 errors
      0 input OLS, 0 LRR, 0 NOS, 0 loop inits
      0 output OLS, 0 LRR, 0 NOS, 0 loop inits
```

Scope of Link Up/Down Trap Settings

The link Up/Down trap settings for the interfaces generate traps based on the following scope:

Switch-level Trap Setting	Interface-level Trap Setting	Trap Generated for Interface Links?
Enabled (defualt)	Enabled (default)	Yes
Enabled	Disabled	No
Disabled	Enabled	No
Disabled	Disabled	No

Displaying SNMP Security Information

Use the **show snmp** commands to display configured SNMP information (see Example 9-1 and 9-6).

Example 9-1 Displays SNMP User Details

```
switch# show snmp user
_____________________________________________________________________
                    SNMP USERS
_____________________________________________________________________

User                            Auth  Priv(enforce) Groups
____                            ____  ______________ ______
admin                           md5   des(no)        network-admin

testusr                         md5   aes-128(no)    role111
                                                     role222
_____________________________________________________________________
 NOTIFICATION TARGET USERS (configured  for sending V3 Inform)
_____________________________________________________________________
User                            Auth  Priv
____                            ____  ____
testtargetusr                   md5   des
(EngineID 0:0:0:63:0:1:0:0:0:15:10:3)
```

Example 9-2 Displays SNMP Community Information

```
switch# show snmp community
Community                       Access
---------                       ------
private                          rw
public                           ro
v93RACqPNH                       ro
```

Example 9-3 Displays SNMP Host Information

```
switch# show snmp host
Host                            Port Version  Level  Type   SecName
____                            ____ _______  ______ ____   _______
171.16.126.34                   2162 v2c      noauth trap   public
171.16.75.106                   2162 v2c      noauth trap   public
...
171.31.58.97                    2162 v2c      auth   trap   public
...
```

The **show snmp** command displays counter information for SNMP contact, location, and packet settings. This command provides information that is used entirely by the Cisco MDS 9000 Family Fabric Manager (refer to the *Cisco Fabric Manager System Management Configuration Guide*). See Example 9-4.

Example 9-4 Displays SNMP Information

```
switch# show snmp
sys contact:
sys location:
1631 SNMP packets input
        0 Bad SNMP versions
        0 Unknown community name
        0 Illegal operation for community name supplied
```

```
        0 Encoding errors
        64294 Number of requested variables
        1 Number of altered variables
        1628 Get-request PDUs
        0 Get-next PDUs
        1 Set-request PDUs
152725 SNMP packets output
        0 Too big errors
        1 No such name errors
        0 Bad values errors
        0 General errors
Community                      Group / Access
---------                      ------
public                          rw

_____________________________________________________________________
                  SNMP USERS
_____________________________________________________________________

User                           Auth  Priv(enforce) Groups
____                           ____  _____________ ______
admin                          md5   des(no)       network-admin

testusr                        md5   aes-128(no)   role111
                                                   role222

_____________________________________________________________________
 NOTIFICATION TARGET USERS (configured  for sending V3 Inform)
_____________________________________________________________________

User                           Auth  Priv
____                           ____  ____
testtargetusr                  md5   des
(EngineID 0:0:0:63:0:1:0:0:0:15:10:3)
```

Example 9-5 Displays SNMP Engine IDs

```
switch# show snmp engineID
Local SNMP engineID: [Hex] 8000000903000DEC2CF180
                     [Dec] 128:000:000:009:003:000:013:236:044:241:128
```

Example 9-6 Displays Information on SNMP Security Groups

```
switch# show snmp group
groupname: network-admin
security model: any
security level: noAuthNoPriv
readview: network-admin-rd
writeview: network-admin-wr
notifyview: network-admin-rd
storage-type: permanent
row status: active

groupname: network-admin
security model: any
security level: authNoPriv
readview: network-admin-rd
writeview: network-admin-wr
notifyview: network-admin-rd
storage-type: permanent
```

```
row status: active

groupname: network-operator
security model: any
security level: noAuthNoPriv
readview: network-operator-rd
writeview: network-operator-wr
notifyview: network-operator-rd
storage-type: permanent
row status: active

groupname: network-operator
security model: any
security level: authNoPriv
readview: network-operator-rd
writeview: network-operator-wr
notifyview: network-operator-rd
storage-type: permanent
row status: active
```

Default Settings

Table 9-2 lists the default settings for all SNMP features in any switch.

Table 9-2 Default SNMP Settings

Parameters	Default
User account	No expiry (unless configured)
Password	None

CHAPTER 10

Configuring Domain Parameters

The Fibre Channel domain (fcdomain) feature performs principal switch selection, domain ID distribution, FC ID allocation, and fabric reconfiguration functions as described in the FC-SW-2 standards. The domains are configured on a per VSAN basis. If you do not configure a domain ID, the local switch uses a random ID.

Caution Changes to fcdomain parameters should not be performed on a daily basis. These changes should be made by an administrator or individual who is completely familiar with switch operations.

Tip When you change the configuration, be sure to save the running configuration. The next time you reboot the switch, the saved configuration is used. If you do not save the configuration, the previously saved startup configuration is used.

This chapter includes the following sections:

Fibre Channel Domains

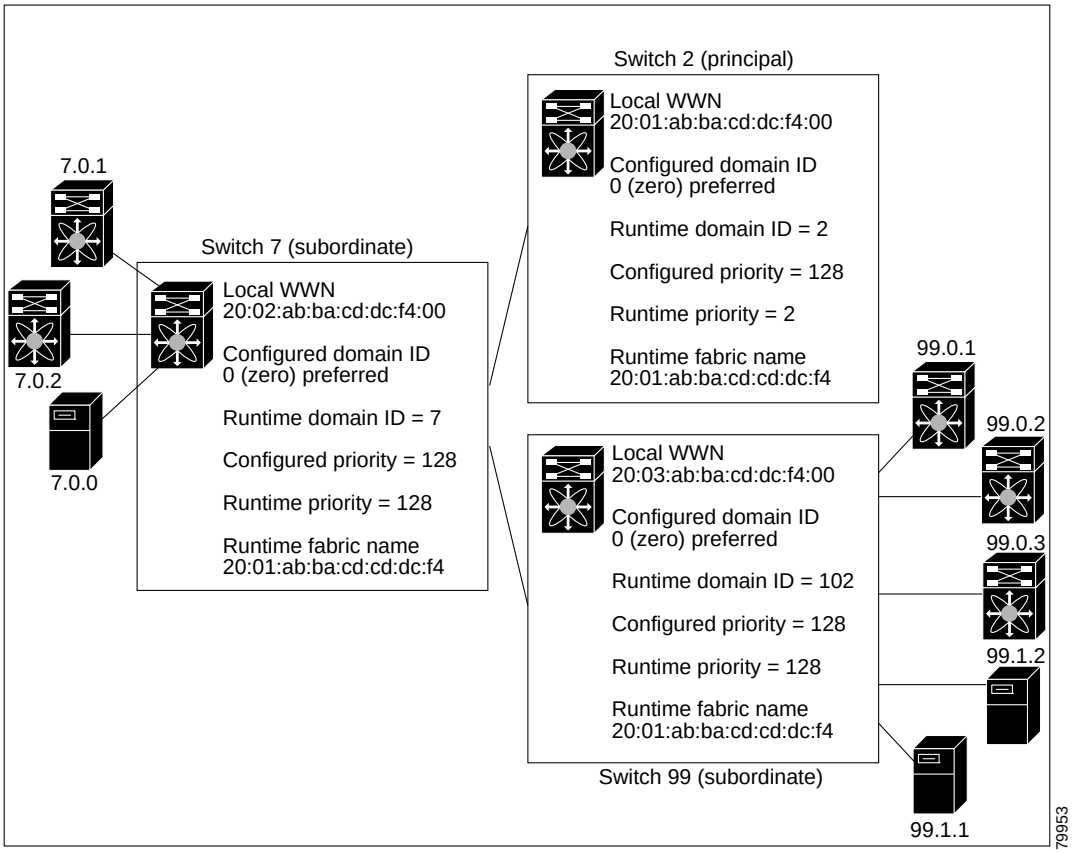
The Fibre Channel domain (fcdomain) feature performs principal switch selection, domain ID distribution, FC ID allocation, and fabric reconfiguration functions as described in the FC-SW-2 standards. The domains are configured on a per VSAN basis. If you do not configure a domain ID, the local switch uses a random ID.

This section describes each fcdomain phase:

- Principal switch selection—This phase guarantees the selection of a unique principal switch across the fabric.
- Domain ID distribution—This phase guarantees each switch in the fabric obtains a unique domain ID.
- FC ID allocation—This phase guarantees a unique FC ID assignment to each device attached to the corresponding switch in the fabric.
- Fabric reconfiguration—This phase guarantees a resynchronization of all switches in the fabric to ensure they simultaneously restart a new principal switch selection phase.

Figure 10-1 shows a sample fcdomain configuration.

Figure 10-1 Sample fcdomain Configuration



Note Domain IDs and VSAN values used in all procedures are only provided as examples. Be sure to use IDs and values that apply to your configuration.

This section describes the fcdomain feature and includes the following topics:

About Domain Restart

Fibre Channel domains can be started disruptively or nondisruptively. If you perform a disruptive restart, reconfigure fabric (RCF) frames are sent to other switches in the fabric and data traffic is disrupted on all the switches in the VSAN (including remotely segmented ISLs). If you perform a nondisruptive restart, build fabric (BF) frames are sent to other switches in the fabric and data traffic is disrupted only on the switch.

If you are attempting to resolve a domain ID conflict, you must manually assign domain IDs. A disruptive restart is required to apply most configuration changes, including manually assigned domain IDs. Nondisruptive domain restarts are acceptable only when changing a preferred domain ID into a static one (and the actual domain ID remains the same).

Note A static domain is specifically configured by the user and may be different from the runtime domain. If the domain IDs are different, the runtime domain ID changes to take on the static domain ID after the next restart, either disruptive or nondisruptive.

Tip If a VSAN is in interop mode, you cannot restart the fcdomain for that VSAN disruptively.

You can apply most of the configurations to their corresponding runtime values. Each of the following sections provide further details on how the fcdomain parameters are applied to the runtime values.

The **fcdomain restart** command applies your changes to the runtime settings. Use the **disruptive** option to apply most of the configurations to their corresponding runtime values, including preferred domain IDs (see the "About Domain IDs" section on page 10-7).

Restarting a Domain

To restart the fabric disruptively or nondisruptively, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain restart vsan 1`	Forces the VSAN to reconfigure without traffic disruption.
	`switch(config)# fcdomain restart disruptive vsan 1`	Forces the VSAN to reconfigure with data traffic disruption.

About Domain Manager Fast Restart

As of Cisco MDS SAN-OS Release 3.0(2), when a principal link fails, the domain manager must select a new principal link. By default, the domain manager starts a build fabric (BF) phase, followed by a principal switch selection phase. Both of these phases involve all the switches in the VSAN and together take at least 15 seconds to complete. To reduce the time required for the domain manager to select a new principal link, you can enable the domain manager fast restart feature.

When fast restart is enabled and a backup link is available, the domain manager needs only a few milliseconds to select a new principal link to replace the one that failed. Also, the reconfiguration required to select the new principal link only affects the two switches that are directly attached to the failed link, not the entire VSAN. When a backup link is not available, the domain manager reverts to the default behavior and starts a BF phase, followed by a principal switch selection phase. The fast restart feature can be used in any interoperability mode.

Tip We recommend using fast restart on most fabrics, especially those with a large number of logical ports (3200 or more), where a logical port is an instance of a physical port in a VSAN.

Enabling Domain Manager Fast Restart

To enable the domain manager fast restart feature in Cisco SAN-OS Release 3.0(2) or later, or MDS NX-OS Release 4.1(1a) or alter, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain optimize fast-restart vsan 3`	Enables domain manager fast restart on VSAN 3.
	`switch(config)# fcdomain optimize fast-restart vsan 7 - 10`	Enables domain manager fast restart on the range of VSANs from VSAN 7 to VSAN 10.
	`switch(config)# no fcdomain optimize fast-restart vsan 8`	Disables (default) domain manager fast restart on VSAN 8.

About Switch Priority

By default, the configured priority is 128. The valid range to set the priority is between 1 and 254. Priority 1 has the highest priority. Value 255 is accepted from other switches, but cannot be locally configured.

Any new switch can become the principal switch when it joins a stable fabric. During the principal switch selection phase, the switch with the highest priority becomes the principal switch. If two switches have the same configured priority, the switch with the lower WWN becomes the principal switch.

The priority configuration is applied to runtime when the fcdomain is restarted (see the "About Domain Restart" section on page 10-3). This configuration is applicable to both disruptive and nondisruptive restarts.

Configuring Switch Priority

To configure the priority for the principal switch, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain priority 25 VSAN 99`	Configures a priority of 25 for the local switch in VSAN 99.
	`switch(config)# no fcdomain priority 25 VSAN 99`	Reverts the priority to the factory default (128) in VSAN 99.

About fcdomain Initiation

By default, the fcdomain feature is enabled on each switch. If you disable the fcdomain feature in a switch, that switch can no longer participate with other switches in the fabric. The fcdomain configuration is applied to runtime through a disruptive restart.

Enabling or Disabling fcdomains

To disable or reenable fcdomains in a single VSAN or a range of VSANs, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# no fcdomain vsan 7-200`	Disables the fcdomain configuration in VSAN 7 through 200.
	`switch(config)# fcdomain vsan 2008`	Enables the fcdomain configuration in VSAN 2008.

Configuring Fabric Names

To set the fabric name value for a disabled fcdomain, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain fabric-name 20:1:ac:16:5e:0:21:01 vsan 3`	Assigns the configured fabric name value in VSAN 3.
	`switch(config)# no fcdomain fabric-name 20:1:ac:16:5e:0:21:01 vsan 3010`	Changes the fabric name value to the factory default (20:01:00:05:30:00:28:df) in VSAN 3010.

About Incoming RCFs

You can configure the rcf-reject option on a per-interface, per-VSAN basis. By default, the rcf-reject option is disabled (that is, RCF request frames are not automatically rejected).

The **rcf-reject** option takes immediate effect takes effect immediately. No fcdomain restart is required.

Rejecting Incoming RCFs

To reject incoming RCF request frames, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# interface fc1/1` `switch(config-if)#`	Configures the specified interface.
Step 3	`switch(config-if)# fcdomain rcf-reject vsan 1`	Enables the RCF filter on the specified interface in VSAN 1.
	`switch(config-if)# no fcdomain rcf-reject vsan 1`	Disables (default) the RCF filter on the specified interface in VSAN 1.

About Autoreconfiguring Merged Fabrics

By default, the autoreconfigure option is disabled. When you join two switches belonging to two different stable fabrics that have overlapping domains, the following cases apply:

- If the autoreconfigure option is enabled on both switches, a disruptive reconfiguration phase is started.
- If the autoreconfigure option is disabled on either or both switches, the links between the two switches become isolated.

The autoreconfigure option takes immediate effect at runtime. You do not need to restart the fcdomain. If a domain is currently isolated due to domain overlap, and you later enable the autoreconfigure option on both switches, the fabric continues to be isolated. If you enabled the autoreconfigure option on both switches before connecting the fabric, a disruptive reconfiguration (RCF) will occur. A disruptive reconfiguration may affect data traffic. You can nondisruptively reconfigure the fcdomain by changing the configured domains on the overlapping links and getting rid of the domain overlap.

Enabling Autoreconfiguration

To enable automatic reconfiguration in a specific VSAN (or range of VSANs), follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain auto-reconfigure vsan 10`	Enables the automatic reconfiguration option in VSAN 10.
	`switch(config)# no fcdomain auto-reconfigure 69`	Disables the automatic reconfiguration option and reverts it to the factory default in VSAN 69.

Domain IDs

Domain IDs uniquely identify a switch in a VSAN. A switch may have different domain IDs in different VSANs. The domain ID is part of the overall FC ID.

This section describes how to configure domain IDs and includes the following topics:

About Domain IDs

The configured domain ID can be preferred or static. By default, the configured domain ID is 0 (zero) and the configured type is preferred.

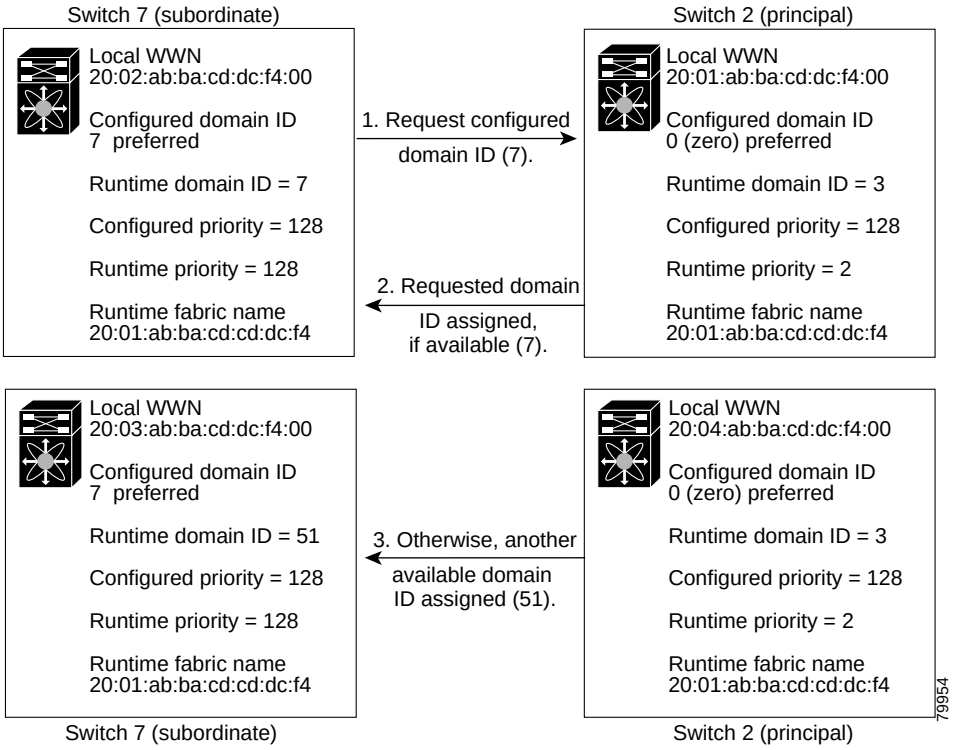
Note The 0 (zero) value can be configured only if you use the preferred option.

If you do not configure a domain ID, the local switch sends a random ID in its request. We recommend that you use static domain IDs.

When a subordinate switch requests a domain, the following process takes place (see Figure 10-2):

1. The local switch sends a configured domain ID request to the principal switch.
2. The principal switch assigns the requested domain ID if available. Otherwise, it assigns another available domain ID.

Figure 10-2 Configuration Process Using the preferred Option



The behavior for a subordinate switch changes based on three factors:

- The allowed domain ID lists.
- The configured domain ID.
- The domain ID that the principal switch has assigned to the requesting switch.

In specific situations, the changes are as follows:

- When the received domain ID is not within the allowed list, the requested domain ID becomes the runtime domain ID and all interfaces on that VSAN are isolated.
- When the assigned and requested domain IDs are the same, the preferred and static options are not relevant, and the assigned domain ID becomes the runtime domain ID.
- When the assigned and requested domain IDs are different, the following cases apply:
  - If the configured type is static, the assigned domain ID is discarded, all local interfaces are isolated, and the local switch assigns itself the configured domain ID, which becomes the runtime domain ID.

– If the configured type is preferred, the local switch accepts the domain ID assigned by the principal switch and the assigned domain ID becomes the runtime domain ID.

If you change the configured domain ID, the change is only accepted if the new domain ID is included in all the allowed domain ID lists currently configured in the VSAN. Alternatively, you can also configure zero-preferred domain ID.

Tip When the FICON feature is enabled in a given VSAN, the domain ID for that VSAN remains in the static state. You can change the static ID value but you cannot change it to the preferred option.

Note In an IVR without NAT configuration, if one VSAN in the IVR topology is configured with static domain IDs, then the other VSANs (edge or transit) in the topology should also be configured with static domain IDs.
In an IVR NAT configuration, if one VSAN in the IVR topology is configured with static domain IDs, then the IVR domains that can be exported to that VSAN must also be assigned static domains.

Caution You must issue the fcdomain restart command if you want to apply the configured domain changes to the runtime domain.

Note If you have configured an allowed domain ID list, the domain IDs that you add must be in that range for the VSAN. See the "About Allowed Domain ID Lists" section on page 10-10.

Specifying Static or Preferred Domain IDs

When you assign a static domain ID type, you are requesting a particular domain ID. If the switch does not get the requested address, it will isolate itself from the fabric. When you specify a preferred domain ID, you are also requesting a particular domain ID; however, if the requested domain ID is unavailable, then the switch will accept another domain ID.

While the static option can be applied at runtime after a disruptive or nondisruptive restart, the preferred option is applied at runtime only after a disruptive restart (see the "About Domain Restart" section on page 10-3).

Note Within a VSAN all switches should have the same domain ID type (either static or preferred). If a configuration is mixed (some switches with static domain types and others with preferred) then you may experience link isolation.

To specify a static or preferred domain ID, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain domain 3 preferred vsan 8`	Configures the switch in VSAN 8 to request a preferred domain ID 3 and accepts any value assigned by the principal switch. The domain is range is 1 to 239.
	`switch(config)# no fcdomain domain 3 preferred vsan 8`	Resets the configured domain ID to 0 (default) in VSAN 8. The configured domain ID becomes 0 preferred.
Step 3	`switch(config)# fcdomain domain 2 static vsan 237`	Configures the switch in VSAN 237 to accept only a specific value and moves the local interfaces in VSAN 237 to an isolated state if the requested domain ID is not granted.
	`switch(config)# no fcdomain domain 18 static vsan 237`	Resets the configured domain ID to factory defaults in VSAN 237. The configured domain ID becomes 0 preferred.

Note When a new domain ID is configured, the new configuration has to be applied by manually restarting the domain using the **fcdomain restart** command; if a discrepancy is detected between the configured domain ID and the runtime domain ID during the subsequent fabric merge, the link will be isolated.

About Allowed Domain ID Lists

By default, the valid range for an assigned domain ID list is from 1 to 239. You can specify a list of ranges to be in the allowed domain ID list and separate each range with a comma. The principal switch assigns domain IDs that are available in the locally configured allowed domain list.

Use allowed domain ID lists to design your VSANs with non-overlapping domain IDs. This helps you in the future if you need to implement IVR without the NAT feature.

Tip If you configure an allowed list on one switch in the fabric, we recommend you configure the same list in all other switches in the fabric to ensure consistency or use CFS to distribute the configuration.

An allowed domain ID list must satisfy the following conditions:

- If this switch is a principal switch, all the currently assigned domain IDs must be in the allowed list.
- If this switch is a subordinate switch, the local runtime domain ID must be in the allowed list.
- The locally configured domain ID of the switch must be in the allowed list.
- The intersection of the assigned domain IDs with other already configured domain ID lists must not be empty.

Configuring Allowed Domain ID Lists

To configure the allowed domain ID list, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain allowed 50-110 vsan 4`	Configures the list to allow switches with the domain ID 50 through 110 in VSAN 4.
	`switch(config)# no fcdomain allowed 50-110 vsan 5`	Reverts to the factory default of allowing domain IDs from 1 through 239 in VSAN 5.

About CFS Distribution of Allowed Domain ID Lists

You can enable the distribution of the allowed domain ID lists configuration information to all Cisco MDS switches in the fabric using the Cisco Fabric Services (CFS) infrastructure. This feature allows you to synchronize the configuration across the fabric from the console of a single MDS switch. Since the same configuration is distributed to the entire VSAN, you avoid possible misconfiguration and the likelihood that two switches in the same VSAN have configured incompatible allowed domains.

Note All switches in the fabric must be running Cisco SAN-OS Release 3.0(1) or later to distribute the allowed domain ID list using CFS.

Use CFS to distribute the allowed domain ID list to ensure consistency in the allowed domain ID lists on all switches in the VSAN.

Note We recommend configuring the allow domain ID list and committing it on the principle switch.

For more information about CFS, see Chapter 2, “Using the CFS Infrastructure”

Enabling Distribution

CFS distribution of allowed domain ID lists is disabled by default. You must enable distribution on all switches to which you want to distribute the allowed domain ID lists.

To enable (or disable) allowed domain ID list configuration distribution, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain distribute`	Enables domain configuration distribution.
	`switch(config)# no fcdomain distribute`	Disables (default) domain configuration distribution.

Locking the Fabric

The first action that modifies the existing configuration creates the pending configuration and locks the feature in the fabric. Once you lock the fabric, the following conditions apply:

- No other user can make any configuration changes to this feature.
- A pending configuration is created by copying the active configuration. Modifications from this point on are made to the pending configuration and remain there until you commit the changes to the active configuration (and other switches in the fabric) or discard them.

Committing Changes

To apply the pending domain configuration changes to other MDS switches in the VSAN, you must commit the changes. The pending configuration changes are distributed and, on a successful commit, the configuration changes are applied to the active configuration in the MDS switches throughout the VSAN and the fabric lock is released.

To commit pending domain configuration changes and release the lock, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain commit vsan 10`	Commits the pending domain configuration changes.

Discarding Changes

At any time, you can discard the pending changes to the domain configuration and release the fabric lock. If you discard (abort) the pending changes, the configuration remains unaffected and the lock is released.

To discard pending domain configuration changes and release the lock, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain abort vsan 10`	Discards the pending domain configuration changes.

Clearing a Fabric Lock

If you have performed a domain configuration task and have not released the lock by either committing or discarding the changes, an administrator can release the lock from any switch in the fabric. If the administrator performs this task, your pending changes are discarded and the fabric lock is released.

Tip The pending changes are only available in the volatile directory and are discarded if the switch is restarted.

To release a fabric lock, issue the **clear fcdomain session vsan** command in EXEC mode using a login ID that has administrative privileges.

```
switch# clear fcdomain session vsan 10
```

Displaying CFS Distribution Status

You can display the status of CFS distribution for allowed domain ID lists using the **show fcdomain status** command.

```
switch# show fcdomain status
CFS distribution is enabled
```

Displaying Pending Changes

You can display the pending configuration changes using the **show fcdomain pending** command:

```
switch# show fcdomain pending vsan 10

Pending Configured Allowed Domains
---------------------------------

VSAN 10
Assigned or unallowed domain IDs: 1-9,24,100,231-239.
[User] configured allowed domain IDs: 10-230.
```

You can display the differences between the pending configuration and the current configuration using the **show fcdomain pending-diff** command.

```
switch# show fcdomain pending-diff vsan 10
Current Configured Allowed Domains
---------------------------------

VSAN 10
Assigned or unallowed domain IDs: 24,100.
[User] configured allowed domain IDs: 1-239.

Pending Configured Allowed Domains
---------------------------------

VSAN 10
Assigned or unallowed domain IDs: 1-9,24,100,231-239.
[User] configured allowed domain IDs: 10-230.
```

Displaying Session Status

You can display the status of the distribution session using the **show fcdomain session-status vsan** command.

```
switch# show fcdomain session-status vsan 1
Last Action: Distribution Enable
Result: Success
```

About Contiguous Domain ID Assignments

By default, the contiguous domain assignment is disabled. When a subordinate switch requests the principal switch for two or more domains and the domains are not contiguous, the following cases apply:

- If the contiguous domain assignment is enabled in the principal switch, the principal switch locates contiguous domains and assigns them to the subordinate switches. If contiguous domains are not available, the NX-OS software rejects this request.
- If the contiguous domain assignment is disabled in the principal switch, the principal switch assigns the available domains to the subordinate switch.

Enabling Contiguous Domain ID Assignments

To enable contiguous domains in a specific VSAN (or a range of VSANs), follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain contiguous-allocation vsan 81-83`	Enables the contiguous allocation option in VSAN 81 through 83. **Note** The **contiguous-allocation** option takes immediate effect at runtime. You do not need to restart the fcdomain.
	`switch(config)# no fcdomain contiguous-allocation vsan 1030`	Disables the contiguous allocation option and reverts it to the factory default in VSAN 1030.

FC IDs

When an N or NL port logs into a Cisco MDS 9000 Family switch, it is assigned an FC ID. By default, the persistent FC ID feature is enabled. If this feature is disabled, the following consequences apply:

- An N or NL port logs into a Cisco MDS 9000 Family switch. The WWN of the requesting N or NL port and the assigned FC ID are retained and stored in a volatile cache. The contents of this volatile cache are not saved across reboots.
- The switch is designed to preserve the binding FC ID to the WWN on a best-effort basis. For example, if one N port disconnects from the switch and its FC ID is requested by another device, this request is granted and the WWN with the initial FC ID association is released.
- The volatile cache stores up to 4000 entries of WWN to FC ID binding. If this cache is full, a new (more recent) entry overwrites the oldest entry in the cache. In this case, the corresponding WWN to FC ID association for the oldest entry is lost.
- The switch connection behavior differs between N ports and NL ports:
  - N ports receive the same FC IDs if disconnected and reconnected to any port within the same switch (as long as it belongs to the same VSAN).
  - NL ports receive the same FC IDs only if connected back to the same port on the switch to which they were originally connected.

This section describes configuring FC IDs and includes the following topics:

About Persistent FC IDs

When persistent FC IDs are enabled, the following consequences apply:

- The currently *in use* FC IDs in the fcdomain are saved across reboots.
- The fcdomain automatically populates the database with dynamic entries that the switch has learned about after a device (host or disk) is plugged into a port interface.

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.

Note FC IDs are enabled by default. This change of default behavior from releases prior to Cisco MDS SAN-OS Release 2.0(1b) prevents FC IDs from being changed after a reboot. You can disable this option for each VSAN.

A persistent FC ID assigned to an F port can be moved across interfaces and can continue to maintain the same persistent FC ID.

Note Persistent FC IDs with loop-attached devices (FL ports) need to remain connected to the same port in which they were configured.

Note Due to differences in Arbitrated Loop Physical Address (ALPA) support on devices, FC ID persistency for loop-attached devices is not guaranteed.

Enabling the Persistent FC ID Feature

To enable the persistent FC ID feature, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.

	Command	Purpose
Step 2	`switch(config)# fcdomain fcid persistent vsan 1000` `FCID(s) persistent feature is enabled.`	Activates (default) persistency of FC IDs in VSAN 1000.
	`switch(config)# no fcdomain fcid persistent vsan 20`	Disables the FC ID persistency feature in VSAN 20.

About Persistent FC ID Configuration

When the persistent FC ID feature is enabled, you can enter the persistent FC ID submode and add static or dynamic entries in the FC ID database. By default, all added entries are static. Persistent FC IDs are configured on a per-VSAN basis. Follow these requirements to manually configure a persistent FC ID:

- Ensure that the persistent FC ID feature is enabled in the required VSAN.
- Ensure that the required VSAN is an active VSAN—persistent FC IDs can only be configured on active VSANs.
- Verify that the domain part of the FC ID is the same as the runtime domain ID in the required VSAN. If the software detects a domain mismatch, the command is rejected.
- Verify that the port field of the FC ID is 0 (zero) when configuring an area.

Note FICON uses a different scheme for allocating FC IDs based in the front panel port number. This scheme takes precedence over FC ID persistence in FICON VSANs.

Configuring Persistent FC IDs

To configure persistent FC IDs, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcdomain fcid database` `switch(config-fcid-db)#`	Enters FC ID database configuration submode.
Step 3	`switch(config-fcid-db)# vsan 1000 wwn 33:e8:00:05:30:00:16:df fcid 0x070128`	Configures a device WWN (33:e8:00:05:30:00:16:df) with the FC ID 0x070128 in VSAN 1000. **Note** To avoid assigning a duplicate FC ID, use the **show fcdomain address-allocation vsan** command to display the FC IDs in use.
	`switch(config-fcid-db)# vsan 1000 wwn 11:22:11:22:33:44:33:44 fcid 0x070123 dynamic`	Configures a device WWN (11:22:11:22:33:44:33:44) with the FC ID 0x070123 in VSAN 1000 in dynamic mode.
	`switch(config-fcid-db)# vsan 1000 wwn 11:22:11:22:33:44:33:44 fcid 0x070100 area`	Configures a device WWN (11:22:11:22:33:44:33:44) with the FC IDs 0x070100 through 0x701FF in VSAN 1000. **Note** To secure the entire area for this fcdomain, assign 00 as the last two characters of the FC ID.

About Unique Area FC IDs for HBAs

Note Only read this section if the HBA port and the storage port are connected to the same switch.

Some HBA ports require a different area ID than storage ports when they are both connected to the same switch. For example, if the storage port FC ID is 0x6f7704, the area for this port is 77. In this case, the HBA port's area can be anything other than 77. The HBA port's FC ID must be manually configured to be different from the storage port's FC ID.

Switches in the Cisco MDS 9000 Family facilitate this requirement with the FC ID persistence feature. You can use this feature to preassign an FC ID with a different area to either the storage port or the HBA port. The procedure in this example uses a switch domain of 111(6f hex). The HBA port connects to interface fc1/9 and the storage port connects to interface fc 1/10 in the same switch.

Configuring Unique Area FC IDs for an HBA

To configure a different area ID for the HBA port, follow these steps:

Step 1 Obtain the Port WWN (Port Name field) ID of the HBA using the **show flogi database** command).

```
switch# show flogi database
--------------------------------------------------------------------------------
```

```
INTERFACE    VSAN    FCID        PORT NAME                    NODE NAME
--------------------------------------------------------------------------------
 fc1/9       3       0x6f7703    50:05:08:b2:00:71:c8:c2      50:05:08:b2:00:71:c8:c0
 fc1/10      3       0x6f7704    50:06:0e:80:03:29:61:0f      50:06:0e:80:03:29:61:0f
```

Note Both FC IDs in this setup have the same area 77 assignment.

Step 2 Shut down the HBA interface in the MDS switch.

```
switch# conf t
switch(config)# interface fc1/9
switch(config-if)# shutdown
switch(config-if)# end
switch#
```

Step 3 Verify that the FC ID feature is enabled using the **show fcdomain vsan** command.

```
switch# show fcdomain vsan 1
...
Local switch configuration information:
        State: Enabled
        FCID persistence: Disabled
```

If this feature is disabled, continue with this procedure to enable the persistent FC ID.

If this feature is already enabled, skip to Step 7.

Step 4 Enable the persistent FC ID feature in the Cisco MDS switch.

```
switch# conf t
switch(config)# fcdomain fcid persistent vsan 1
switch(config)# end
switch#
```

Step 5 Assign a new FC ID with a different area allocation. In this example, we replace *77* with *ee*.

```
switch# conf t
switch(config)# fcdomain fcid database
switch(config-fcid-db)# vsan 3 wwn 50:05:08:b2:00:71:c8:c2 fcid 0x6fee00 area
```

Step 6 Enable the HBA interface in the Cisco MDS switch.

```
switch# conf t
switch(config)# interface fc1/9
switch(config-if)# no shutdown
switch(config-if)# end
switch#
```

Step 7 Verify the pWWN ID of the HBA using the **show flogi database** command.

```
switch# show flogi database
--------------------------------------------------------------------------------
INTERFACE    VSAN    FCID        PORT NAME                    NODE NAME
--------------------------------------------------------------------------------
fc1/9        3       0x6fee00    50:05:08:b2:00:71:c8:c2      50:05:08:b2:00:71:c8:c0
fc1/10       3       0x6f7704    50:06:0e:80:03:29:61:0f      50:06:0e:80:03:29:61:0f
```

Note Both FC IDs now have different area assignments.

About Persistent FC ID Selective Purging

Persistent FC IDs can be purged selectively. Static entries and FC IDs currently in use cannot be deleted. Table 10-1 identifies the FC ID entries that are deleted or retained when persistent FC IDs are purged.

Table 10-1 Purged FC IDs

Persistent FC ID state	Persistent Usage State	Action
Static	In use	Not deleted
Static	Not in use	Not deleted
Dynamic	In use	Not deleted
Dynamic	Not in use	Deleted

Purging Persistent FC IDs

To purge persistent FC IDs, follow this step:

	Command	Purpose
Step 1	`switch# purge fcdomain fcid vsan 4`	Purges all dynamic and unused FC IDs in VSAN 4.
	`switch# purge fcdomain fcid vsan 3-5`	Purges dynamic and unused FC IDs in VSAN 3, 4, and 5.

Displaying fcdomain Information

Use the **show fcdomain** command to display global information about fcdomain configurations. See Example 10-1.

Note In Example 10-1, the fcdomain feature is disabled. Consequently, the runtime fabric name is the same as the configured fabric name.

Example 10-1 Displays the Global fcdomain Information

```
switch# show fcdomain vsan 2
The local switch is the Principal Switch.

Local switch run time information:
        State: Stable
        Local switch WWN:    20:01:00:0b:46:79:ef:41
        Running fabric name: 20:01:00:0b:46:79:ef:41
        Running priority: 128
        Current domain ID: 0xed(237)

Local switch configuration information:
        State: Enabled
        FCID persistence: Disabled
        Auto-reconfiguration: Disabled
        Contiguous-allocation: Disabled
        Configured fabric name: 20:01:00:05:30:00:28:df
        Configured priority: 128
        Configured domain ID: 0x00(0) (preferred)
```

```
Principal switch run time information:
        Running priority: 128

No interfaces available.
```

Use the **show fcdomain domain-list** command to display the list of domain IDs of all switches belonging to a specified VSAN. This list provides the WWN of the switches owning each domain ID. Example 10-2 shows the following:

- A switch with WWN of 20:01:00:05:30:00:47:df is the principal switch and has domain 200.
- A switch with WWN of 20:01:00:0d:ec:08:60:c1 is the local switch (the one where you typed the CLI command to show the domain-list) and has domain 99.
- The IVR manager obtained virtual domain 97 using 20:01:00:05:30:00:47:df as the WWN for a virtual switch.

Example 10-2 Displays the fcdomain Lists

```
switch# show fcdomain domain-list vsan 76

Number of domains: 3
Domain ID              WWN
---------    -----------------------
0xc8(200)    20:01:00:05:30:00:47:df [Principal]
 0x63(99)    20:01:00:0d:ec:08:60:c1 [Local]
 0x61(97)    50:00:53:0f:ff:f0:10:06 [Virtual (IVR)]
```

Use the **show fcdomain allowed vsan** command to display the list of allowed domain IDs configured on this switch. See Example 10-3.

Example 10-3 Displays the Allowed Domain ID Lists

```
switch# show fcdomain allowed vsan 1
Assigned or unallowed domain IDs: 1-96,100,111-239.
[Interoperability Mode 1] allowed domain IDs: 97-127.
[User] configured allowed domain IDs: 50-110.
```

Tip Ensure that the requested domain ID passes the Cisco NX-OS software checks, if **interop 1** mode is required in this switch.

Use the **show fcdomain fcid persistent** command to display all existing, persistent FC IDs for a specified VSAN. You can also specify the **unused** option to view only persistent FC IDs that are still not in use. See Examples 10-4 and 10-5.

Example 10-4 Displays Persistent FC IDs in a Specified VSAN

```
switch# show fcdomain fcid persistent vsan 1000
Total entries 2.

Persistent FCIDs table contents:
VSAN             WWN                FCID        Mask        Used    Assignment
----    -----------------------    --------    -----------    ----    ----------
1000    11:11:22:22:11:11:12:23    0x700101    SINGLE FCID      NO    STATIC
1000    44:44:33:33:22:22:11:11    0x701000    ENTIRE AREA      NO    DYNAMIC
```

Example 10-5 Displays All Persistent FC IDs in the fcdomain

```
switch# show fcdomain fcid persistent
Total entries 2.

Persistent FCIDs table contents:
VSAN             WWN                FCID         Mask         Used    Assignment
----    ----------------------    --------    -----------    ----    ----------
1000    11:11:22:22:11:11:22:22    0x700501    SINGLE FCID      NO    STATIC
1003    44:44:33:33:22:22:11:11    0x781000    ENTIRE AREA     YES    DYNAMIC
```

Use the **show fcdomain statistics** command to display frame and other fcdomain statistics for a specified VSAN or PortChannel. See Example 10-6 and Example 10-7.

Example 10-6 Displays fcdomain Statistics for a Specified VSAN

```
switch# show fcdomain statistics vsan 1
VSAN Statistics
        Number of Principal Switch Selections: 5
        Number of times Local Switch was Principal: 0
        Number of 'Build Fabric's: 3
        Number of 'Fabric Reconfigurations': 0
```

Example 10-7 Displays fcdomain Statistics for a Specified PortChannel

```
switch# show fcdomain statistics interface port-channel 10 vsan 1
        Interface Statistics:
                Transmitted      Received
                -----------      --------
                EFPs      13       9
                DIAs      7        7
                RDIs      0        0
                ACCs      21       25
                RJTs      1        1
                BFs       2        2
                RCFs      4        4
                Error     0        0
                Total     48       48
        Total Retries: 0
        Total Frames: 96
                -----------      --------
```

Use the **show fcdomain address-allocation** command to display FC ID allocation statistics including a list of assigned and free FC IDs. See Example 10-8.

Example 10-8 Displays FC ID Information

```
switch# show fcdomain address-allocation vsan 1
Free FCIDs: 0x020000 to 0x02fdff
            0x02ff00 to 0x02fffe

Assigned FCIDs: 0x02fe00 to 0x02feff
                0x02ffff

Reserved FCIDs: 0x020100 to 0x02f0ff
                0x02fe00 to 0x02feff
                0x02ffff


Number free FCIDs: 65279
Number assigned FCIDs: 257
```

```
Number reserved FCIDs: 61697
```

Use the **show fcdomain address-allocation cache** command to display the valid address allocation cache. The cache is used by the principal switch to reassign the FC IDs for a device (disk or host) that exited and reentered the fabric. In the cache content, VSAN refers to the VSAN that contains the device, WWN refers to the device that owned the FC IDs, and mask refers to a single or entire area of FC IDs. See Example 10-9.

Example 10-9 Displays Address Allocation Information

```
switch# show fcdomain address-allocation cache
Cache content:
line#     VSAN              WWN                FCID         mask
-----     ----    ------------------------    --------    -----------
   1.       12    21:00:00:e0:8b:08:a2:21    0xef0400    ENTIRE AREA
   2.        6    50:06:04:82:c3:a1:2f:5c    0xef0002    SINGLE FCID
   3.        8    20:4e:00:05:30:00:24:5e    0xef0300    ENTIRE AREA
   4.        8    50:06:04:82:c3:a1:2f:52    0xef0001    SINGLE FCID
```

Default Settings

Table 10-2 lists the default settings for all fcdomain parameters.

Table 10-2 Default fcdomain Parameters

Parameters	Default
fcdomain feature	Enabled.
Configured domain ID	0 (zero).
Configured domain	Preferred.
auto-reconfigure option	Disabled.
contiguous-allocation option	Disabled.
Priority	128.
Allowed list	1 to 239.
Fabric name	20:01:00:05:30:00:28:df.
rcf-reject	Disabled.
Persistent FC ID	Enabled.
Allowed domain ID list configuration distribution	Disabled.

CHAPTER 11

Monitoring Network Traffic Using SPAN

This chapter describes the Switched Port Analyzer (SPAN) features provided in switches in the Cisco MDS 9000 Family.

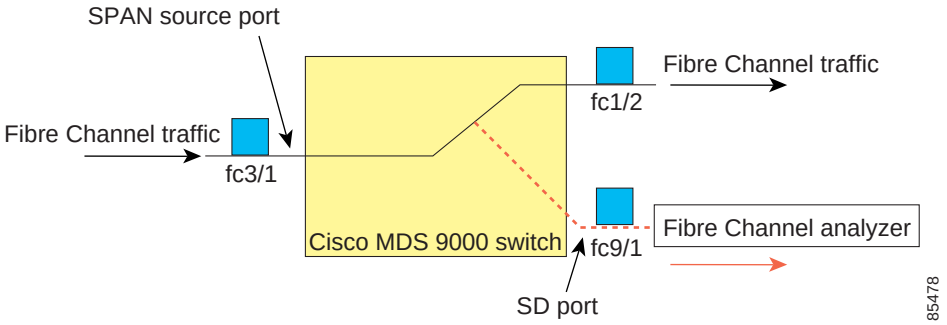
It includes the following sections:

About SPAN

The SPAN feature is specific to switches in the Cisco MDS 9000 Family. It monitors network traffic through a Fibre Channel interface. Traffic through any Fibre Channel interface can be replicated to a special port called the SPAN destination port (SD port). Any Fibre Channel port in a switch can be configured as an SD port. Once an interface is in SD port mode, it cannot be used for normal data traffic. You can attach a Fibre Channel Analyzer to the SD port to monitor SPAN traffic.

SD ports do not receive frames, they only transmit a copy of the SPAN source traffic. The SPAN feature is non-intrusive and does not affect switching of network traffic for any SPAN source ports (see Figure 11-1).

Figure 11-1 SPAN Transmission

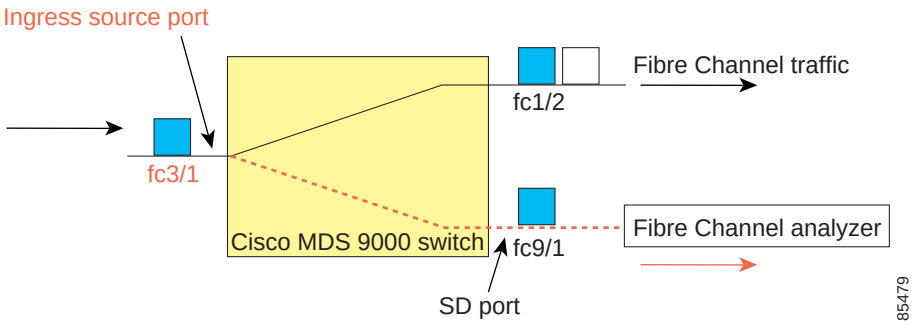


SPAN Sources

SPAN sources refer to the interfaces from which traffic can be monitored. You can also specify VSAN as a SPAN source, in which case, all supported interfaces in the specified VSAN are included as SPAN sources. When a VSAN as a source is specified, then all physical ports and PortChannels in that VSAN are included as SPAN sources. You can choose the SPAN traffic in the ingress direction, the egress direction, or both directions for any source interface:

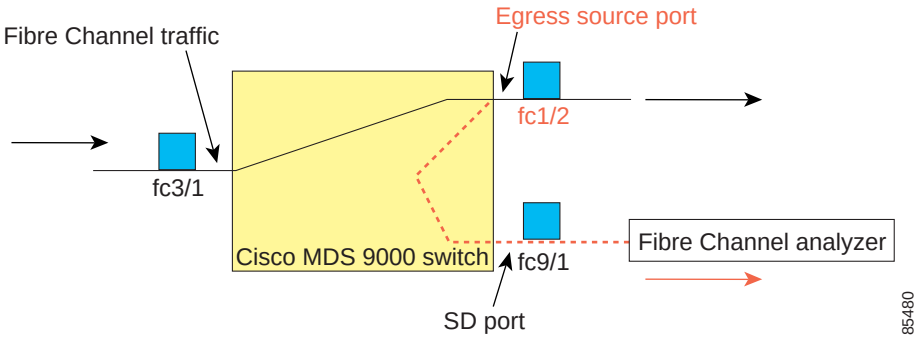
- Ingress source (Rx)—Traffic entering the switch fabric through this source interface is *spanned* or copied to the SD port (see Figure 11-2).

Figure 11-2 SPAN Traffic from the Ingress Direction



- Egress source (Tx)—Traffic exiting the switch fabric through this source interface is spanned or copied to the SD port (see Figure 11-3).

Figure 11-3 SPAN Traffic from Egress Direction



IPS Source Ports

SPAN capabilities are available on the IP Storage Services (IPS) module. The SPAN feature is only implemented on the FCIP and iSCSI virtual Fibre Channel port interfaces, not the physical Gigabit Ethernet ports. You can configure SPAN for ingress traffic, egress traffic, or traffic in both directions for all eight iSCSI and 24 FCIP interfaces that are available in the IPS module.

Note You can configure SPAN for Ethernet traffic using Cisco switches or routers connected to the Cisco MDS 9000 Family IPS modules.

Allowed Source Interface Types

The SPAN feature is available for the following interface types:

- Physical ports such as F ports, FL ports, TE ports, E ports, and TL ports.
- Interface sup-fc0 (traffic to and from the supervisor):
  - The Fibre Channel traffic from the supervisor module to the switch fabric through the sup-fc0 interface is called ingress traffic. It is spanned when sup-fc0 is chosen as an ingress source port.
  - The Fibre Channel traffic from the switch fabric to the supervisor module through the sup-fc0 interface is called egress traffic. It is spanned when sup-fc0 is chosen as an egress source port.
- PortChannels
  - All ports in the PortChannel are included and spanned as sources.
  - You cannot specify individual ports in a PortChannel as SPAN sources. Previously configured SPAN-specific interface information is discarded.
- IPS module specific Fibre Channel interfaces:
  - iSCSI interfaces
  - FCIP interfaces

VSAN as a Source

SPAN sources refer to the interfaces from which traffic can be monitored. When a VSAN as a source is specified, then all physical ports and PortChannels in that VSAN are included as SPAN sources. A TE port is included only when the port VSAN of the TE port matches the source VSAN. A TE port is excluded even if the configured allowed VSAN list may have the source VSAN, but the port VSAN is different.

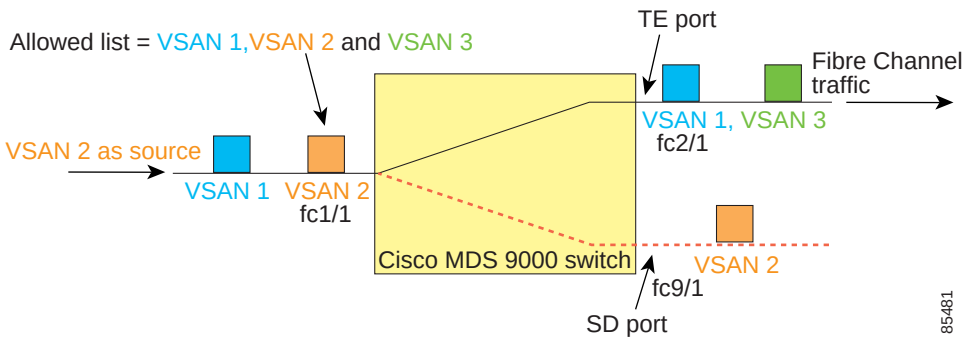
You cannot configure source interfaces (physical interfaces, PortChannels, or sup-fc interfaces) and source VSANs in the same SPAN session.

Guidelines to Configure VSANs as a Source

The following guidelines apply when configuring VSANs as a source:

- Traffic on all interfaces included in a source VSAN is spanned only in the ingress direction.
- If a VSAN is specified as a source, you cannot perform interface-level SPAN configuration on the interfaces that are included in the VSAN. Previously configured SPAN-specific interface information is discarded.
- If an interface in a VSAN is configured as a source, you cannot configure that VSAN as a source. You must first remove the existing SPAN configurations on such interfaces before configuring VSAN as a source.
- Interfaces are only included as sources when the port VSAN matches the source VSAN. Figure 11-4 displays a configuration using VSAN 2 as a source:
  - All ports in the switch are in VSAN 1 except fc1/1.
  - Interface fc1/1 is the TE port with port VSAN 2. VSANs 1, 2, and 3 are configured in the allowed list.
  - VSAN 1 and VSAN 2 are configured as SPAN sources.

Figure 11-4 VSAN as a Source



For this configuration, the following apply:

- VSAN 2 as a source includes only the TE port fc1/1 that has port VSAN 2.
- VSAN 1 as a source does not include the TE port fc1/1 because the port VSAN does not match VSAN 1.

SPAN Sessions

Each SPAN session represents an association of one destination with a set of source(s) along with various other parameters that you specify to monitor the network traffic. One destination can be used by one or more SPAN sessions. You can configure up to 16 SPAN sessions in a switch. Each session can have several source ports and one destination port.

To activate any SPAN session, at least one source and the SD port must be up and functioning. Otherwise, traffic is not directed to the SD port.

Tip A source can be shared by two sessions, however, each session must be in a different direction—one ingress and one egress.

You can temporarily deactivate (suspend) any SPAN session. The traffic monitoring is stopped during this time.

Specifying Filters

You can perform VSAN-based filtering to selectively monitor network traffic on specified VSANs. You can apply this VSAN filter to all sources in a session (see Figure 11-4). Only VSANs present in the filter are spanned.

You can specify session VSAN filters that are applied to all sources in the specified session. These filters are bidirectional and apply to all sources configured in the session. Each SPAN session represents an association of one destination with a set of source(s) along with various other parameters that you specify to monitor the network traffic.

Guidelines to Specifying Filters

The following guidelines apply to SPAN filters:

- PortChannel configurations are applied to all ports in the PortChannel.
- If no filters are specified, the traffic from all active VSANs for that interface is spanned by default.
- While you can specify arbitrary VSAN filters in a session, traffic can only be monitored on the port VSAN or on allowed-active VSANs in that interface.

SD Port Characteristics

An SD port has the following characteristics:

- Ignores BB_credits.
- Allows data traffic only in the egress (Tx) direction.
- Does not require a device or an analyzer to be physically connected.
- Supports only 1 Gbps or 2 Gbps speeds. The auto speed option is not allowed.
- Multiple sessions can share the same destination ports.
- If the SD port is shut down, all shared sessions stop generating SPAN traffic.

- The outgoing frames can be encapsulated in Extended Inter-Switch Link (EISL) format.
- The SD port does not have a port VSAN.
- SD ports cannot be configured using Storage Services Modules (SSMs).
- The port mode cannot be changed if it is being used for a SPAN session.

Note If you need to change an SD port mode to another port mode, first remove the SD port from all sessions and then change the port mode using the **switchport mode** command.

Guidelines to Configure SPAN

The following guidelines apply for SPAN configurations:

- You can configure up to 16 SPAN sessions with multiple ingress (Rx) sources.
- You can configure a maximum of three SPAN sessions with one egress (Tx) port.
- In a 32-port switching module, you must configure the same session in all four ports in one port group (unit). If you wish, you can also configure only two or three ports in this unit.
- SPAN frames are dropped if the sum of the bandwidth of the sources exceeds the speed of the destination port.
- Frames dropped by a source port are not spanned.

To configure an SD port for SPAN monitoring, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# interface fc9/1`	Configures the specified interface.
Step 3	`switch(config-if)# switchport mode SD`	Configures the SD port mode for interface fc9/1.
Step 4	`switch(config-if)# switchport speed 1000`	Configures the SD port speed to 1000 Mbps.
Step 5	`switch(config-if)# no shutdown`	Enables traffic flow through this interface.

To configure a SPAN session, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span session 1` `switch(config-span)#`	Configures the specified SPAN session (1). If the session does not exist, it is created.
	`switch(config)# no span session 1`	Deletes the specified SPAN session (1).
Step 3	`switch(config-span)# destination interface fc9/1`	Configures the specified destination interface (fc 9/1) in a session.
	`switch(config-span)# no destination interface fc9/1`	Removes the specified destination interface (fc 9/1).

	Command	Purpose
Step 4	`switch(config-span)# source interface fc7/1`	Configures the source (fc7/1) interface in both directions. **Note** The Cisco MDS 9124 Fabric Switch does not support bi-directional SPAN sessions (Rx and Tx)
	`switch(config-span)# no source interface fc7/1`	Removes the specified destination interface (fc 7/1) from this session.
Step 5	`switch(config-span)# source interface sup-fc0`	Configures the source interface (sup-fc0) in the session.
	`switch(config-span)# source interface fc1/5 - 6, fc2/1 -3`	Configures the specified interface ranges in the session.
	`switch(config-span)# source vsan 1-2`	Configures source VSANs 1 and 2 in the session.
	`switch(config-span)# source interface port-channel 1`	Configures the source PortChannel (port-channel 1).
	`switch(config-span)# source interface fcip 51`	Configures the source FCIP interface in the session.
	`switch(config-span)# source interface iscsi 4/1`	Configures the source iSCSI interface in the session.
	`switch(config-span)# source interface svc1/1 tx traffic-type initiator`	Configures the source SVC interface in the Tx direction for an initiator traffic type.
	`switch(config-span)# no source interface port-channel 1`	Deletes the specified source interface (port-channel 1).

To configure a SPAN filter, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span session 1` `switch(config-span)#`	Configures the specified session (1).
Step 3	`switch(config-span)# source interface fc9/1 tx`	Configures the source fc9/1 interface in the egress (Tx) direction.
	`switch(config-span)# source filter vsan 1-2`	Configures VSANs 1 and 2 as session filters.
	`switch(config-span)# source interface fc7/1 rx`	Configures the source fc7/1 interface in the ingress (Rx) direction.

Configuring SPAN max-queued-packets

When a SPAN destination port is oversubscribed or has more source traffic than the speed of the destination port, the source ports of the SPAN session will reduce in their throughput. The impact is proportional to the amount of source traffic flowing in. Lowering the max-queued-packets value from the default value of 15 to 1 prevents the impact on the source ports. It is necessary to reconsider the default value for this setting as it may impact the source interface throughput.

To configure SPAN max-queued-packets for a SPAN session, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span max-queued-packtes 1` `switch(config-span)#`	Configures the SPAN max-queued-packets.

By default, SPAN frames are dropped if the sum of the bandwidth of the source interfaces exceed the bandwidth of the destination port. With a higher value, the SPAN traffic has a higher probability of reaching the SPAN destination port instead of being dropped at the expense of data traffic throughput.

Note The span max-queued-packets can be changed only if no SPAN sessions are currently active on the switch.

Note If you are spanning the traffic going through an FCIP interface, SPAN copies may be dropped even if the SD interface has more bandwidth than the amount of traffic being replicated. To avoid SPAN drops, set the max-queued-packets to a higher value; for example, 100.

Configuring SPAN for Generation 2 Fabric Switches

Cisco Generation 2 fabric switches (such as MDS 9124) support SPAN sessions in both directions, Rx and Tx.

Note While using Generation 2 fabric switches, you cannot create an additional active SPAN session when you already have one.

The following examples show how to configure SPAN sessions for the ingress and egress direction for this switch.

Example 11-1 Configuring a Generation 2 Fabric Switch for Ingress SPAN sessions

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span session 1` `switch(config-span)#`	Configures the specified session (1).
Step 3	`switch(config-span)# destination interface fc1/1`	Configures interface fc1/1 as the destination.
Step 4	`switch(config-span)# source interface fc1/2 rx`	Configures the source interface fc1/2 in the ingress direction.

Example 11-2 Configuring a Generation 2 Fabric Switch for Egress SPAN Session

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span session 1` `switch(config-span)#`	Configures the specified session (1).
Step 3	`switch(config-span)# destination interface fc1/1`	Configures interface fc1/1 as the destination.
Step 4	`switch(config-span)# source interface fc1/2 tx`	Configures the source interface fc1/2 in the egress direction.

You can specify multiple SPAN source interfaces in Rx and Tx directions. However, the direction should be explicitly mentioned at the end of the command. The SPAN will reject any source interface configuration that fails to mention the direction.

Example 11-3 Configuring Cisco MDS 9124 for Multiple SPAN Interfaces

```
switch(config-span)# span session 1
switch(config-span)# destination interface fc1/1
switch(config-span)# source interface fc1/2-6 rx
switch(config-span)# source interface fc1/2-6 tx
```

Generation 2 Fabric Switches support VSAN filters for one VSAN only in the egress direction; this restriction does not apply to the ingress direction. For example, if you have an interface that is a TE port, with an active VSAN of 1 to 5, and you specify a VSAN filter for VSAN 2, then only the traffic on VSAN 2 will be filtered.

```
switch(config-span)# span session 1
switch(config-span)# source filter vsan 2
switch(config-span)# destination interface fc1/1
switch(config-span)# source interface fc1/2 tx
```

However, if you specify the VSAN filter for VSANs 1 to 2, then traffic from all VSANs (1 to 5) is filtered, which makes the filter useless.

```
switch(config-span)# span session 1
switch(config-span)# source filter vsan 1-2
switch(config-span)# destination interface fc1/1
switch(config-span)# source interface fc1/2 tx
```

Suspending and Reactivating SPAN Sessions

You can temporarily deactivate (suspend) any SPAN session. The traffic monitoring is stopped during this time.

To temporarily suspend or reactivate a SPAN session filter, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span session 1` `switch(config-span)#`	Configures the specified session (1).
Step 3	`switch(config-span)# suspend`	Temporarily suspends the session.
	`switch(config-span)# no suspend`	Reactivates the session.

Encapsulating Frames

The frame encapsulation feature is disabled by default. If you enable the encapsulation feature, all outgoing frames are encapsulated.

The **switchport encap eisl** command only applies to SD port interfaces. If encapsulation is enabled, you see a new line (`Encapsulation is eisl`) in the **show interface** *SD_port_interface* command output.

To encapsulate outgoing frames (optional), follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# interface fc9/32`	Configures the specified interface.
Step 3	`switch(config-if)# switchport mode SD`	Configures the SD port mode for interface fc9/32.
Step 4	`switch(config-if)# switchport encap eisl`	Enables the encapsulation option for this SD port.
	`switch(config-if)# no switchport encap eisl`	Disables (default) the encapsulation option.

SPAN Conversion Behavior

SPAN features (configured in any prior release) are converted as follows:

- If source interfaces and source VSANs are configured in a given session, then all the source VSANs are removed from that session.

  For example, before Cisco MDS SAN-OS Release 1.0(4):

```
Session 1 (active)
   Destination is fc1/9
   No session filters configured
   Ingress (rx) sources are
      vsans 10-11
     fc1/3,
   Egress (tx) sources are
     fc1/3,
```

  Once upgraded to Cisco MDS SAN-OS Release 1.1(1):

```
Session 1 (active)
   Destination is fc1/9
   No session filters configured
   Ingress (rx) sources are
     fc1/3,
   Egress (tx) sources are
     fc1/3,
```

  Session 1 had both source interfaces and source VSANs before the upgrade. After the upgrade, the source VSANs were removed (rule 1).

- If interface level VSAN filters are configured in source interfaces, then the source interfaces are also removed from the session. If this interface is configured in both directions, it is removed from both directions.

  For example, before Cisco MDS SAN-OS Release 1.0(4):

```
Session 2 (active)
   Destination is fc1/9
   No session filters configured
   Ingress (rx) sources are
      vsans 12
```

```
    fc1/6 (vsan 1-20),
  Egress (tx) sources are
    fc1/6 (vsan 1-20),
```

Once upgraded to Cisco MDS SAN-OS Release 1.1(1):

```
Session 2 (inactive as no active sources)
   Destination is fc1/9
   No session filters configured
   No ingress (rx) sources
   No egress (tx) sources
```

Note The deprecated configurations are removed from persistent memory once a switchover or a new startup configuration is implemented.

Session 2 had a source VSAN 12 and a source interface fc1/6 with VSAN filters specified in Cisco MDS SAN-OS Release 1.0(4). When upgraded to Cisco MDS SAN-OS Release 1.1(1) the following changes are made:

- The source VSAN (VSAN 12) is removed (rule 1).
- The source interface fc1/6 had VSAN filters specified—it is also removed (rule 2).

Monitoring Traffic Using Fibre Channel Analyzers

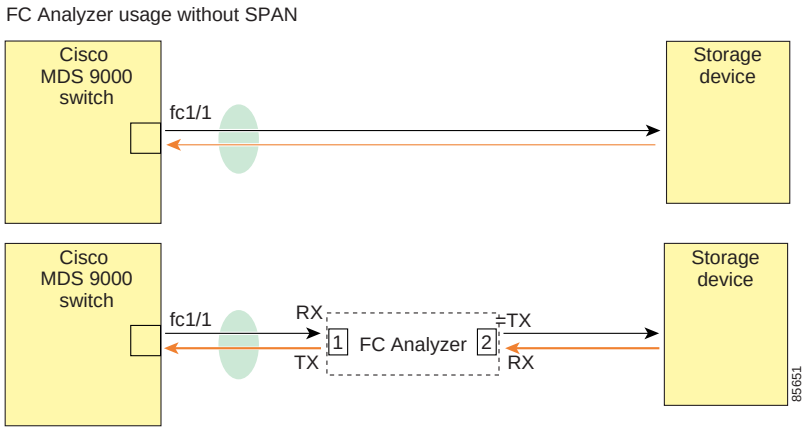
You can use SPAN to monitor traffic on an interface without any traffic disruption. This feature is especially useful in troubleshooting scenarios in which traffic disruption changes the problem environment and makes it difficult to reproduce the problem. You can monitor traffic in either of the following two ways:

- Without SPAN
- With SPAN

Without SPAN

You can monitor traffic using interface fc1/1 in a Cisco MDS 9000 Family switch that is connected to another switch or host. You need to physically connect a Fibre Channel analyzer between the switch and the storage device to analyze the traffic through interface fc1/1 (see Figure 11-5).

Figure 11-5 Fibre Channel Analyzer Usage Without SPAN



This type of connection has the following limitations:

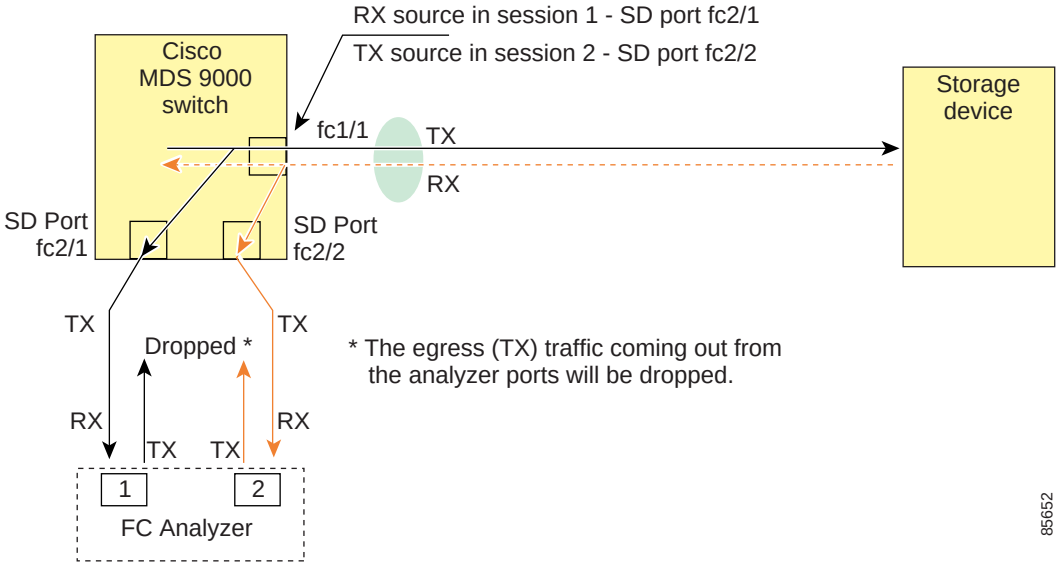
- It requires you to physically insert the FC analyzer between the two network devices.
- It disrupts traffic when the Fibre Channel analyzer is physically connected.
- The analyzer captures data only on the Rx links in both port 1 and port 2. Port 1 captures traffic exiting interface fc1/1 and port 2 captures ingress traffic into interface fc1/1.

With SPAN

Using SPAN you can capture the same traffic scenario (see Figure 11-5) without any traffic disruption. The Fibre Channel analyzer uses the ingress (Rx) link at port 1 to capture all the frames going out of the interface fc1/1. It uses the ingress link at port 2 to capture all the ingress traffic on interface fc1/1.

Using SPAN you can monitor ingress traffic on fc1/1 at SD port fc2/2 and egress traffic on SD port fc2/1. This traffic is seamlessly captured by the FC analyzer (see Figure 11-6).

Figure 11-6 Fibre Channel Analyzer Using SPAN



Configuring Fibre Channel Analyzers Using SPAN

To configure Fibre Channel Analyzers using SPAN for the example in Figure 11-6, follow these steps:

Step 1 Configure SPAN on interface fc1/1 in the ingress (Rx) direction to send traffic on SD port fc2/1 using session 1.

Step 2 Configure SPAN on interface fc1/1in the egress (Tx) direction to send traffic on SD port fc2/2 using session 2.

Step 3 Physically connect fc2/1 to port 1 on the Fibre Channel analyzer.

Step 4 Physically connect fc2/2 to port 2 on the Fibre Channel analyzer.

To configure SPAN on the source and destination interfaces, follow these steps:

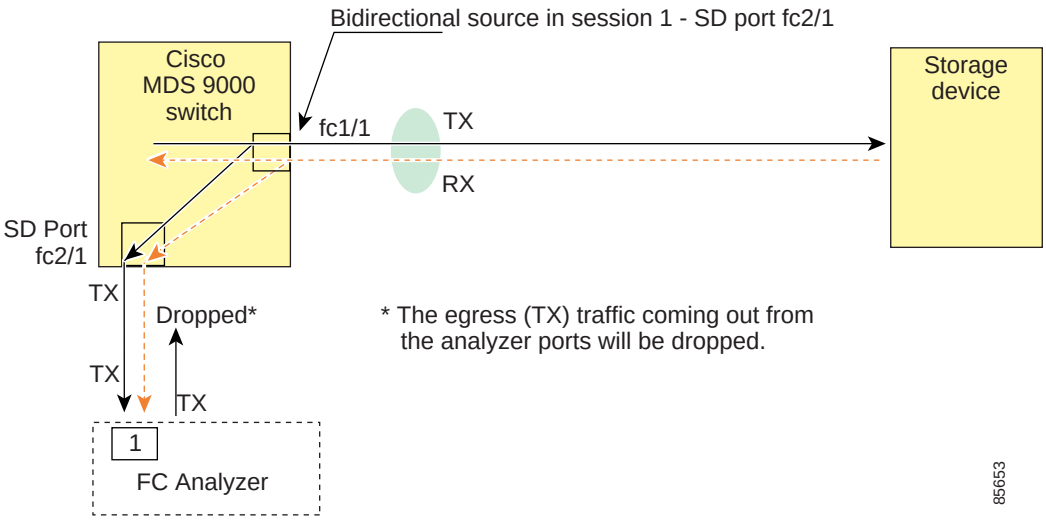
	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span session 1` `switch(config-span)#`	Creates the SPAN session 1.
Step 3	`switch(config-span)## destination interface fc2/1`	Configures the destination interface fc2/1.
Step 4	`switch(config-span)# source interface fc1/1 rx`	Configures the source interface fc1/1 in the ingress direction.
Step 5	`switch(config)# span session 2` `switch(config-span)#`	Creates the SPAN session 2.
Step 6	`switch(config-span)## destination interface fc2/2`	Configures the destination interface fc2/2.
Step 7	`switch(config-span)# source interface fc1/1 tx`	Configures the source interface fc1/1 in the egress direction.

Single SD Port to Monitor Traffic

You do not need to use two SD ports to monitor bidirectional traffic on any interface (see Figure 11-6). You can use one SD port and one FC analyzer port by monitoring traffic on the interface at the same SD port fc2/1.

Figure 11-7 shows a SPAN setup where one session with destination port fc2/1 and source interface fc1/1 is used to capture traffic in both ingress and egress directions. This setup is more advantageous and cost effective than the setup shown in Figure 11-6—it uses one SD port and one port on the analyzer, instead of using a full, two-port analyzer.

Figure 11-7 Fibre Channel Analyzer Using a Single SD Port



To use this setup, the analyzer should have the capability of distinguishing ingress and egress traffic for all captured frames.

To configure SPAN on a single SD port, follow these steps:

	Command	Purpose
Step 1	`switch# config t`	Enters configuration mode.
Step 2	`switch(config)# span session 1` `switch(config-span)#`	Creates the SPAN session 1.
Step 3	`switch(config-span)## destination interface fc2/1`	Configures the destination interface fc2/1.
Step 4	`switch(config-span)# source interface fc1/1`	Configures the source interface fc1/1 on the same SD port.

Displaying SPAN Information

Use the **show span** command to display configured SPAN information. See Examples 11-4 to 11-9.

Example 11-4 Displays SPAN Sessions in a Brief Format

```
switch# show span session brief
```

```
-------------------------------------------------------
 Session  Admin          Oper           Destination
          State          State          Interface
-------------------------------------------------------
 7        no suspend     active         fc2/7
 1        suspend        inactive       not configured
 2        no suspend     inactive       fc3/1
```

Example 11-5 Displays a Specific SPAN Session in Detail

```
switch# show span session 7
Session 7 (active)
   Destination is fc2/7
   No session filters configured
   No ingress (rx) sources
   Egress (tx) sources are
     port-channel 7,
```

Example 11-6 Displays ALL SPAN Sessions

```
switch# show span session
Session 1 (inactive as no destination)
Destination is not specified
   Session filter vsans are 1
   No ingress (rx) sources
   No egress (tx) sources
Session 2 (active)
   Destination is fc9/5
   No session filters configured
   Ingress (rx) sources are
      vsans 1
   No egress (tx) sources
Session 3 (admin suspended)
   Destination is not configured
   Session filter vsans are 1-20
   Ingress (rx) sources are
     fc3/2, fc3/3, fc3/4, fcip 51,
     port-channel 2, sup-fc0,
   Egress (tx) sources are
     fc3/2, fc3/3, fc3/4, sup-fc0,
```

Example 11-7 Displays SPAN drop-counters for the SPAN Sessions

```
switch# show span drop-counters
SPAN Drop-Counters for module 3 is: 0x0
SPAN Drop-Counters for module 7 is: 0x0
```

Note The **show span drop-counters** command displays the dropped counters. You can configure the **show span max-queued-packets** command only if the dropped counter value is greater than zero.

Example 11-8 Displays SPAN max-queued-packets for the SPAN Sessions

```
switch# show span max-queued-packets
Drop-Threshold for SPAN sessions: 15
```

Note The default value for all the SPAN sessions is 15. The **span max-queued-packets** command can be issued only if the sessions are inactive.

Example 11-9 Displays an SD Port Interface with Encapsulation Enabled

```
switch# show int fc9/32
fc9/32 is up
    Hardware is Fibre Channel
    Port WWN is 22:20:00:05:30:00:49:5e
    Admin port mode is SD
    Port mode is SD
    Port vsan is 1
    Speed is 1 Gbps
    Receive Buffer Size is 2112
    Encapsulation is eisl <---------- Displays the enabled encapsulation status
    Beacon is turned off
    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 frames input, 0 bytes, 0 discards
        0 CRC,  0 unknown class
        0 too long, 0 too short
      0 frames output, 0 bytes, 0 discards
      0 input OLS, 0 LRR, 0 NOS, 0 loop inits
      0 output OLS, 0 LRR, 0 NOS, 0 loop inits
```

Remote SPAN

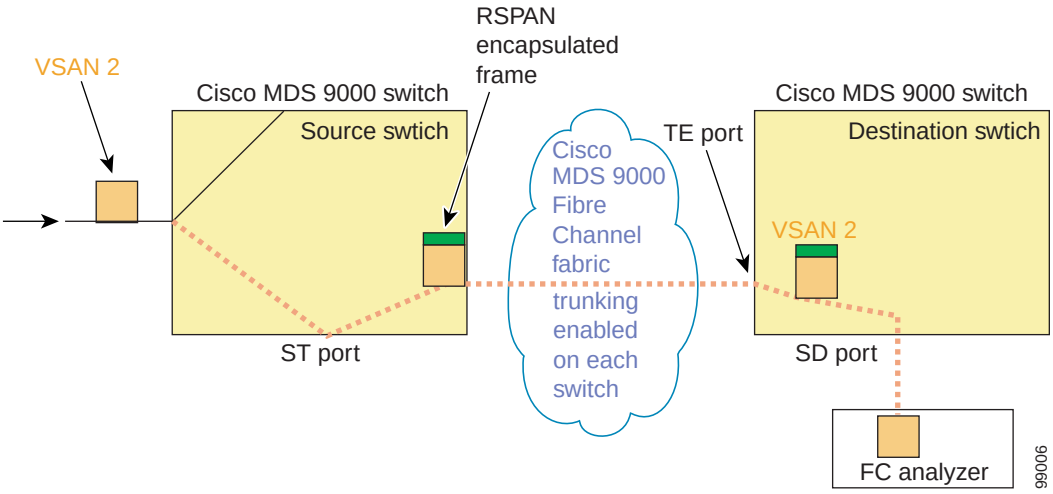
Note Remote SPAN is not supported on the Cisco Fabric Switch for HP c-Class BladeSystem and the Cisco Fabric Switch for IBM BladeSystem.

The Remote SPAN (RSPAN) feature enables you to remotely monitor traffic for one or more SPAN sources distributed in one or more source switches in a Fibre Channel fabric. The SPAN destination (SD) port is used for remote monitoring in a destination switch. A destination switch is usually different from the source switch(es) but is attached to the same Fibre Channel fabric. You can replicate and monitor traffic in any remote Cisco MDS 9000 Family switch or director, just as you would monitor traffic in a Cisco MDS source switch.

The RSPAN feature is nonintrusive and does not affect network traffic switching for those SPAN source ports. Traffic captured on the remote switch is tunneled across a Fibre Channel fabric which has trunking enabled on all switches in the path from the source switch to the destination switch. The Fibre Channel tunnel is structured using trunked ISL (TE) ports. In addition to TE ports, the RSPAN feature uses two other interface types (see Figure 11-8):

- SD ports—A passive port from which remote SPAN traffic can be obtained by the FC analyzer.
- ST ports—A SPAN tunnel (ST) port is an entry point port in the source switch for the RSPAN Fibre Channel tunnel. ST ports are special RSPAN ports and cannot be used for normal Fibre Channel traffic.

Figure 11-8 RSPAN Transmission



Advantages to Using RSPAN

The RSPAN features has the following advantages:

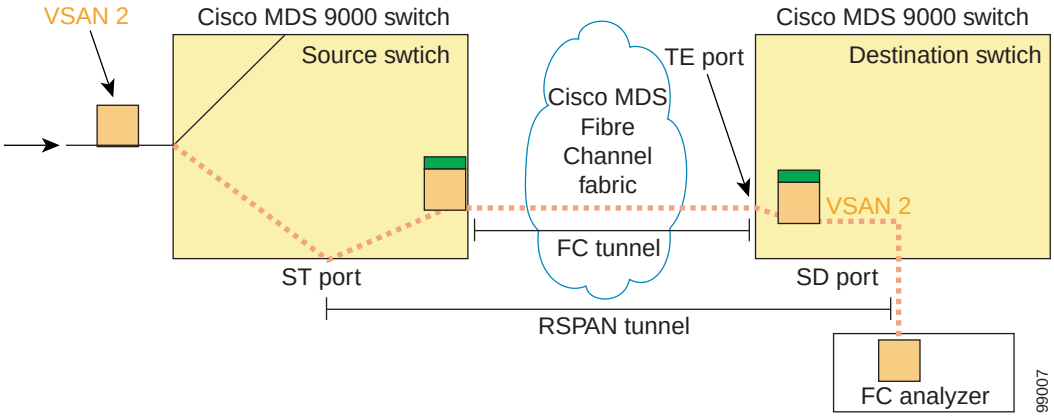
- Enables nondisruptive traffic monitoring at a remote location.
- Provides a cost effective solution by using one SD port to monitor remote traffic on multiple switches.
- Works with any Fibre Channel analyzer.
- Is compatible with the Cisco MDS 9000 Port Analyzer adapters.
- Does not affect traffic in the source switch, but shares the ISL bandwidth with other ports in the fabric.

FC and RSPAN Tunnels

An FC tunnel is a logical data path between a source switch and a destination switch. The FC tunnel originates from the source switch and terminates at the remotely located destination switch.

RSPAN uses a special Fibre Channel tunnel (FC tunnel) that originates at the ST port in the source switch and terminates at the SD port in the destination switch. You must bind the FC tunnel to an ST port in the source switch and map the same FC tunnel to an SD port in the destination switch. Once the mapping and binding is configured, the FC tunnel is referred to as an RSPAN tunnel (see Figure 11-9).

Figure 11-9 FC and RSPAN Tunnel



RSPAN Configuration Guidelines

The following guidelines apply for a SPAN configuration:

- All switches in the end-to-end path of the RSPAN tunnel must belong to the Cisco MDS 9000 Family.
- All VSANs with RSPAN traffic must be enabled. If a VSAN containing RSPAN traffic is not enabled, it is dropped.
- The following configurations must be performed on *each* switch in the end-to-end path of the Fibre Channel tunnel in which RSPAN is to be implemented:
  - Trunking must be enabled (enabled by default) and the trunk enabled link must be the lowest cost link in the path.
  - VSAN interface must be configured.
  - The Fibre Channel tunnel feature must be enabled (disabled by default).
  - IP routing must be enabled (disabled by default).

Note If the IP address is in the same subnet as the VSAN, the VSAN interface does not have to be configured for all VSANs on which the traffic is spanned.

- A single Fibre Channel switch port must be dedicated for the ST port functionality.
- Do not configure the port to be monitored as the ST port.
- The FC tunnel's IP address must reside in the same subnet as the VSAN interface.

ST Port Characteristics

ST ports have the following characteristics:

- ST ports perform the RSPAN encapsulation of the FC frame.
- ST ports do not use BB_credits.
- One ST port can only be bound to one FC tunnel.

- ST ports cannot be used for any purpose other than to carry RSPAN traffic.
- ST ports cannot be configured using Storage Services Modules (SSMs).

RSPAN Configuration Example

This section provides a RSPAN configuration example using the procedure defined in the previous section.

Configuration in the Source Switch

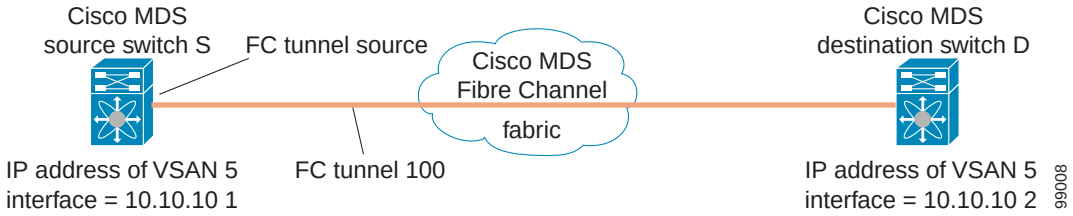
This section identifies the tasks that must be performed in the source switch (Switch S):

- Creating VSAN Interfaces, page 11-19
- Enabling FC Tunnels, page 11-20
- Initiating the FC Tunnel, page 11-20
- Configuring the ST Port, page 11-21
- Configuring an RSPAN Session, page 11-21
- Configuring VSAN Interfaces, page 11-22
- Enabling FC Tunnels, page 11-22
- Enabling IP Routing, page 11-23
- Configuring VSAN Interfaces, page 11-23
- Enabling FC Tunnels, page 11-20
- Configuring the SD Port, page 11-23
- Mapping the FC Tunnel, page 11-24

Creating VSAN Interfaces

Figure 11-10 depicts a basic FC tunnel configuration.

Figure 11-10 FC Tunnel Configuration



Note This example assumes that VSAN 5 is already configured in the VSAN database.

To create a VSAN interface in the source switch for the scenario in Figure 11-10, follow these steps:

	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# interface vsan 5` `switchS(config-if)#`	Configures the specified VSAN interface (VSAN 5) in the source switch (switch S).
Step 3	`switchS(config-if)# ip address 10.10.10.1 255.255.255.0`	Configures the IPv4 address and subnet for the VSAN interface 5 in the source switch (switch S).
Step 4	`switchS(config-if)# no shutdown`	Enables traffic flow through this interface.

Enabling FC Tunnels

To enable the FC tunnel feature, follow these steps:

	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# fc-tunnel enable`	Enables the FC tunnel feature (disabled by default).

Note Be sure to enable this feature in each switch in the end-to-end path in the fabric.

Initiating the FC Tunnel

To initiate the FC tunnel in the source switch for the scenario in Figure 11-10, follow these steps:

	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# interface fc-tunnel 100` `switchS(config-if)#`	Initiates the FC tunnel (100) in the source switch (switch S). The tunnel IDs range from 1 to 255.
Step 3	`switchS(config-if)# source 10.10.10.1`	Maps the IPv4 address of the source switch (switch S) to the FC tunnel (100).
Step 4	`switchS(config-if)# destination 10.10.10.2`	Maps the IPv4 address of the destination switch (switch D) to the FC tunnel (100).
Step 5	`switchS(config-if)# no shutdown`	Enables traffic flow through this interface.

Caution FC Tunnels do not work over non-trunking ISLs.

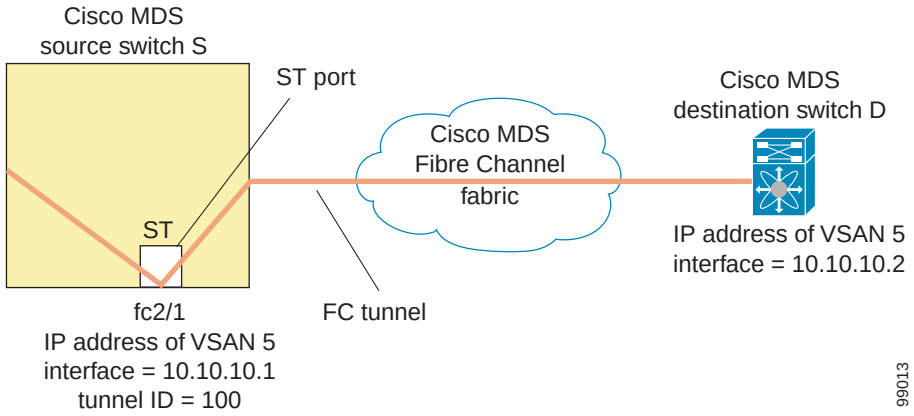
Tip The interface cannot be operationally up until the FC tunnel mapping is configured in the destination switch.

Configuring the ST Port

Once the FC tunnel is created, be sure to configure the ST port to bind it to the FC tunnel at the source switch. The FC tunnel becomes an RSPAN tunnel once the binding and mapping is complete.

Figure 11-11 depicts a basic FC tunnel configuration.

Figure 11-11 Binding the FC Tunnel



Note ST ports cannot be configured using Storage Services Modules (SSMs).

To configure an ST port for the scenario in Figure 11-11, follow these steps:

	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# interface fc2/1`	Configures the specified interface.
Step 3	`switchS(config-if)# switchport mode ST`	Configures the ST port mode for interface fc2/1.
Step 4	`switchS(config-if)# switchport speed 2000`	Configures the ST port speed to 2000 Mbps.
Step 5	`switchS(config-if)# rspan-tunnel interface fc-tunnel 100`	Associates and binds the ST port with the RSPAN tunnel (100).
Step 6	`switchS(config-if)# no shutdown`	Enables traffic flow through this interface.

Configuring an RSPAN Session

A RSPAN session is similar to a SPAN session, with the destination interface being an RSPAN tunnel.

To configure an RSPAN session in the source switch for the scenario in Figure 11-11, follow these steps:

	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# span session 2` `switchS(config-span)#`	Configures the specified SPAN session (2). If the session does not exist, it is created. The session ID ranges from 1 to 16.

	Command	Purpose
Step 3	`switchS(config-span)# destination interface fc-tunnel 100`	Configures the specified RSPAN tunnel (100) in a session.
Step 4	`switchS(config-span)# source interface fc1/1`	Configures the source interface (fc1/1) for this session and spans the traffic from interface fc1/1 to RSPAN tunnel 100.

Configuration in All Intermediate Switches

This section identifies the tasks that must be performed in all intermediate switches in the end-to-end path of the RSPAN tunnel:

- Configuring VSAN Interfaces, page 11-22
- Enabling FC Tunnels, page 11-22
- Enabling IP Routing, page 11-23

Configuring VSAN Interfaces

Figure 11-12 on page 11-23 depicts an RSPAN tunnel configuration terminating in the destination switch (Switch D).

Note This example assumes that VSAN 5 is already configured in the VSAN database.

To create a VSAN interface in the destination switch for the scenario in Figure 11-12 on page 11-23, follow these steps:

	Command	Purpose
Step 1	`switchD# config t`	Enters configuration mode.
Step 2	`switchD(config)# interface vsan 5` `switchD(config-if)#`	Configures the specified VSAN interface (VSAN 5) in the destination switch (Switch D).
Step 3	`switchD(config-if)# ip address 10.10.10.2 255.255.255.0`	Configures the IPv4 address and subnet for the VSAN interface in the destination switch (Switch D).
Step 4	`switchD(config-if)# no shutdown`	Enables traffic flow to administratively allow traffic (provided the operational state is up).

Enabling FC Tunnels

To enable the FC tunnel feature, follow these steps:

	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# fc-tunnel enable`	Initiates the FC tunnel (100) in the source switch (switch S). The tunnel IDs range from 1 to 255.

Note Be sure to enable this feature in each switch in the end-to-end path in the fabric.

Enabling IP Routing

The IP routing feature is disabled by default. Be sure to enable IP routing in each switch (including the source and destination switches) in the end-to-end path in the fabric. This step is required to set up the FC tunnel.

Configuration in the Destination Switch

This section identifies the tasks that must be performed in the destination switch (Switch D):

- Configuring VSAN Interfaces, page 11-23
- Configuring the SD Port, page 11-23
- Mapping the FC Tunnel, page 11-24

Configuring VSAN Interfaces

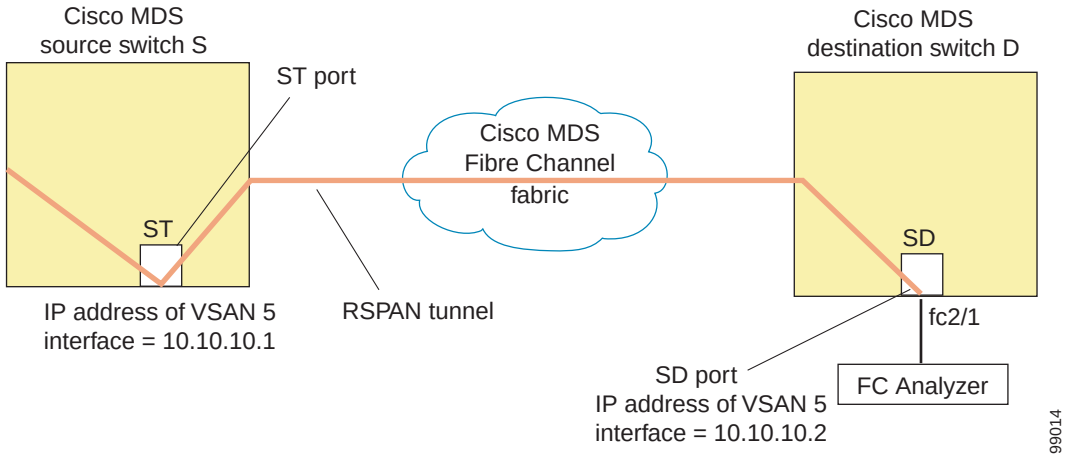
Figure 11-12 on page 11-23 depicts an RSPAN tunnel configuration terminating in the destination switch (Switch D).

Note This example assumes that VSAN 5 is already configured in the VSAN database.

Configuring the SD Port

The SD port in the destination switch enables the FC analyzer to receive the RSPAN traffic from the Fibre Channel tunnel. Figure 11-12 depicts an RSPAN tunnel configuration, now that tunnel destination is also configured.

Figure 11-12 RSPAN Tunnel Configuration



Note SD ports cannot be configured using Storage Services Modules (SSMs).

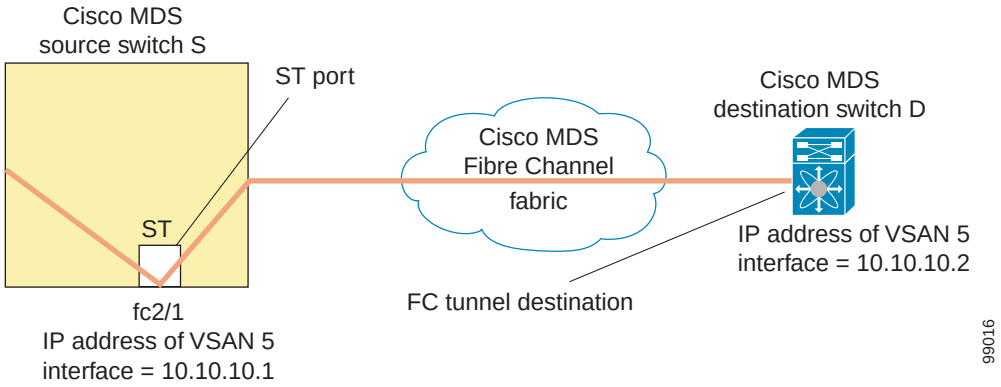
To configure an SD port for the scenario in Figure 11-12, follow these steps:

	Command	Purpose
Step 1	`switchD# config t`	Enters configuration mode.
Step 2	`switchD(config)# interface fc2/1`	Configures the specified interface.
Step 3	`switchD(config-if)# switchport mode SD`	Configures the SD port mode for interface fc2/1.
Step 4	`switchD(config-if)# switchport speed 2000`	Configures the SD port speed to 2000 Mbps.
Step 5	`switchD(config-if)# no shutdown`	Enables traffic flow through this interface.

Mapping the FC Tunnel

The **tunnel-id-map** option specifies the egress interface of the tunnel at the destination switch (see Figure 11-13).

Figure 11-13 FC Tunnel Configuration



To terminate the FC tunnel in the destination switch for the scenario in Figure 11-13, follow these steps:

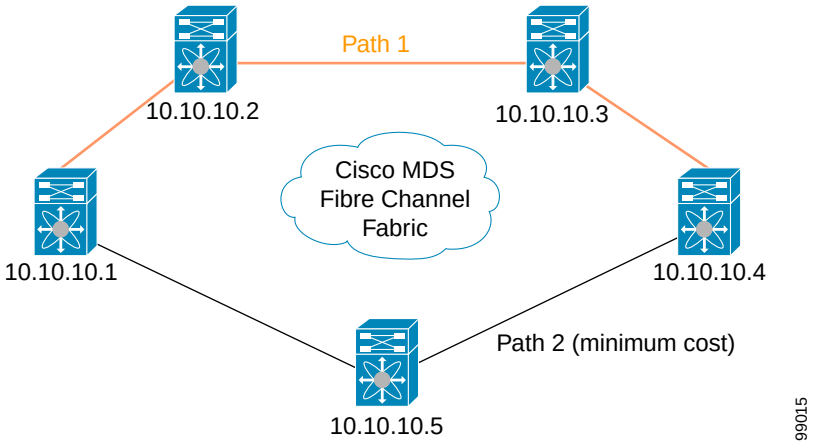
	Command	Purpose
Step 1	`switchD# config t`	Enters configuration mode.
Step 2	`switchD(config)# fc-tunnel tunnel-id-map 100 interface fc2/1`	Terminates the FC tunnel (100) in the destination switch (switch D). The tunnel ID range is from 1 to 255.

Explicit Paths

You can specify an explicit path through the Cisco MDS Fibre Channel fabric (source-based routing), using the **explicit-path** option. For example, if you have multiple paths to a tunnel destination, you can use this option to specify the FC tunnel to always take one path to the destination switch. The software then uses this specified path even if other paths are available.

This option is especially useful if you prefer to direct the traffic through a certain path although other paths are available. In an RSPAN situation, you can specify the explicit path so the RSPAN traffic does not interfere with the existing user traffic. You can create any number of explicit paths in a switch (see Figure 11-14).

Figure 11-14 Explicit Path Configuration



The explicit path must be created in the source switch. To configure an explicit path, you must first create the path and then configure the use of any one path. If an explicit path is not configured, the minimum cost path is used by default. If an explicit path is configured and is functioning, the specified path is used.

To create an explicit path for the scenario in Figure 11-14, follow these steps:

	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# fc-tunnel explicit-path Path1` `switch(config-explicit-path)#`	Places you at the explicit path prompt for the path named Path 1.

Step 3	`switchS(config-explicit-path)# next-address 10.10.10.2 strict`
	`switchS(config-explicit-path)# next-address 10.10.10.3 strict`
	`switchS(config-explicit-path)# next-address 10.10.10.4 strict`
		Specifies that the next hop VSAN interface IPv4 addresses and the previous hops specified in the explicit path do not require direct connection.

Step 4	`switchS(config)# fc-tunnel explicit-path Path2` `switch(config-explicit-path)#`	Places you at the explicit path prompt for Path2.
Step 5	`switchS(config-explicit-path)# next-address 10.10.10.5 strict` `switchS(config-explicit-path)# next-address 10.10.10.4 strict`	Specifies that the next hop VSAN interface IPv4 addresses and the previous hops specified in the explicit path do not require direct connection.
Step 6	`switchS(config)# fc-tunnel explicit-path Path3` `switch(config-explicit-path)#`	Places you at the explicit path prompt for Path3.
Step 7	`switchS(config-explicit-path)# next-address 10.10.10.3 loose`	Configures a minimum cost path in which the 10.10.10.3 IPv4 address exists. **Note** In Figure 11-14, Path 3 is the same as Path 1—10.10.10.3 exists in Path 1. Using the **loose** option, you can achieve the same effect with one command instead of issuing three commands (using the **strict** option) in Step 3.

To reference the explicit path, follow these steps:

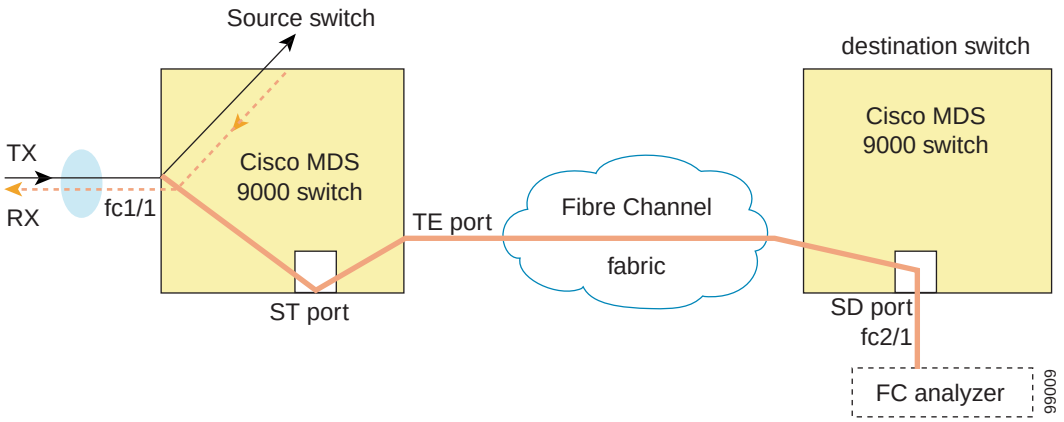
	Command	Purpose
Step 1	`switchS# config t`	Enters configuration mode.
Step 2	`switchS(config)# interface fc-tunnel 100`	References the tunnel ID for Path1.
Step 3	`switchS(config)# explicit-path Path1`	Links Path1 to the tunnel ID.

This configuration explicitly specifies Path 1 to be used for the RSPAN traffic. Refer to RFC 3209 for further details on explicit paths and source-based routing.

Monitoring RSPAN Traffic

Once the session is configured, other SPAN sources for this session can also be configured as required. Figure 11-15 shows an RSPAN setup where one session with destination port fc2/1 and source interface fc1/1 is used to capture traffic in both ingress and egress directions.

Figure 11-15 Fibre Channel Analyzer Using a Single SD Port to Monitor RSPAN Traffic



To use this setup, the analyzer should have the capability of distinguishing ingress and egress traffic for all captured frames.

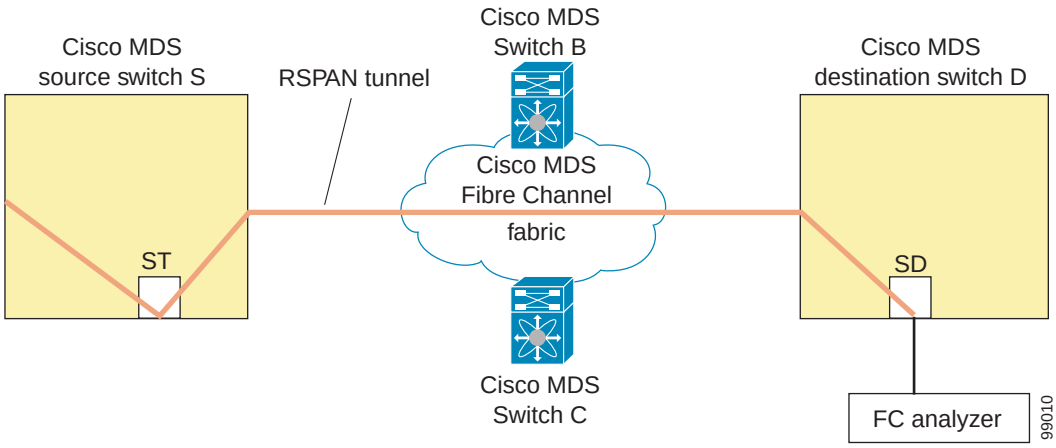
Sample Scenarios

Note RSPAN can be combined with the local SPAN feature so SD ports forward local SPAN traffic along with remote SPAN traffic. Various SPAN source and tunnel scenarios are described in this section.

Single Source with One RSPAN Tunnel

The source Switch S and the destination Switch D are interconnected through a Fibre Channel fabric. An RSPAN tunnel is configured as a destination interface for the SPAN session and the ST port forwards SPAN traffic through the RSPAN tunnel (see Figure 11-16).

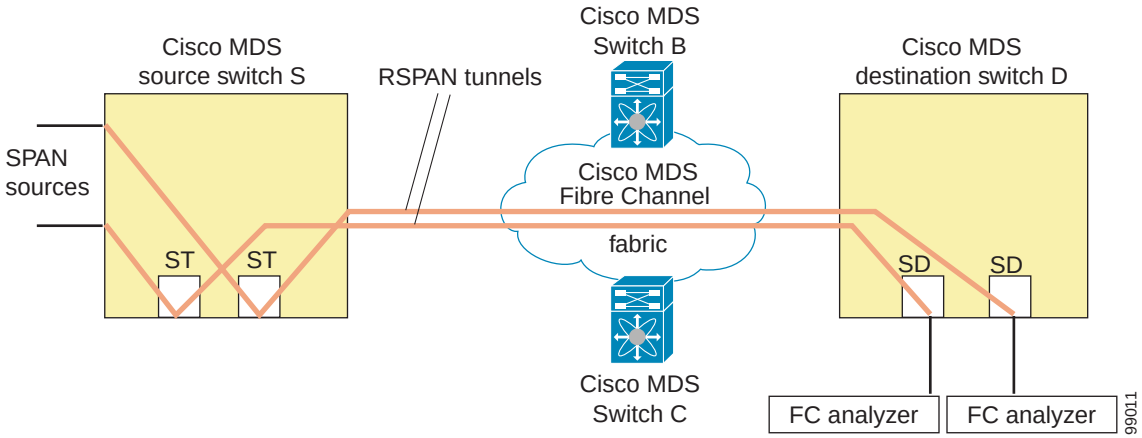
Figure 11-16 RSPAN Scenario with One Source Switch, One Destination Switch, and One Tunnel



Single Source with Multiple RSPAN Tunnels

Figure 11-17 displays two separate RSPAN tunnels configured between Switches S and N. Each tunnel has an associated ST port in the source switch and a separate SD port in the destination switch. This configuration is useful for troubleshooting purposes.

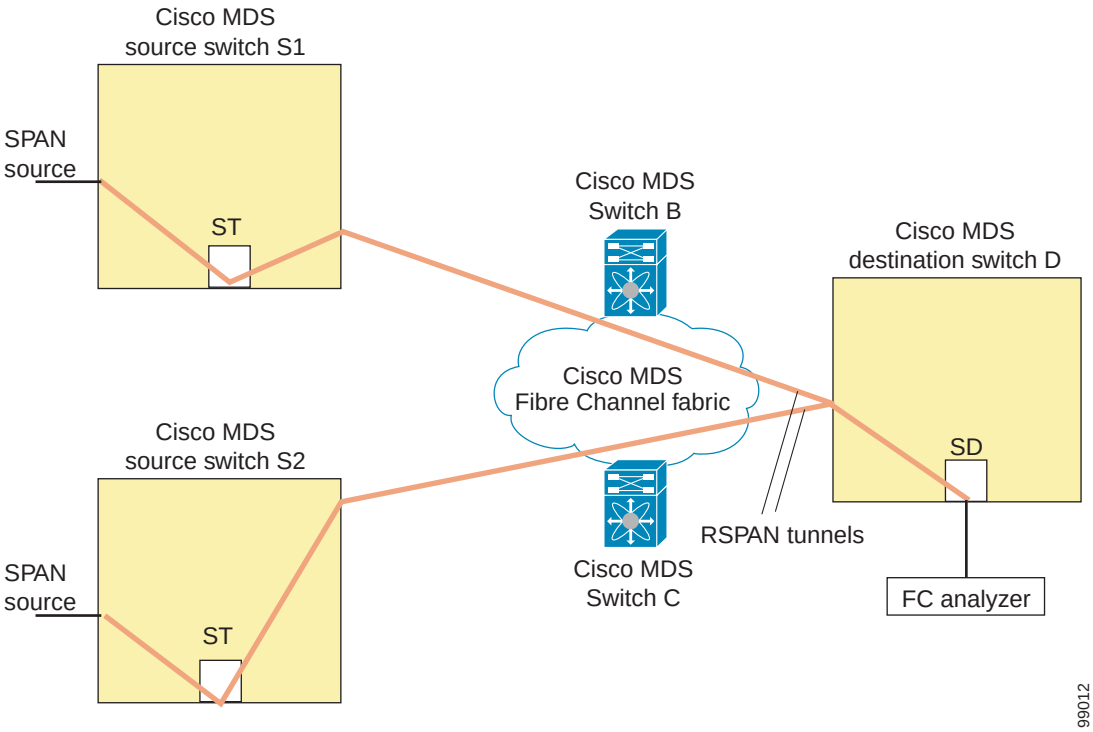
Figure 11-17 RSPAN Scenario with One Source Switch, One Destination Switch, and Multiple Tunnels



Multiple Sources with Multiple RSPAN Tunnels

Figure 11-18 displays two separate RSPAN tunnels configured between Switches S1 and S2. Both tunnels have an associated ST port in their respective source switch and terminate in the same SD port in the destination switch.

Figure 11-18 RSPAN Scenario with Two Source Switches, a Destination Switch, and Multiple Tunnels



This configuration is useful for remote monitoring purposes. For example, the administrator may be at the destination switch and can remotely monitor the two source switches.

Displaying RSPAN Information

Use the **show** commands to display configured RSPAN information. See Examples 11-10 to 11-16.

Example 11-10 Displays ST Port Interface Information

```
switch# show interface brief

--------------------------------------------------------------------------------
Interface   Vsan    Admin   Admin   Status      Oper    Oper    Port-channel
                    Mode    Trunk               Mode    Speed
                            Mode                        (Gbps)
--------------------------------------------------------------------------------
fc1/1       1       auto    on      trunking    TE          2       --
...
fc1/14      1       auto    on      trunking    TE          2       --
fc1/15      1       ST      on      up          ST          2       --
...
fc2/9       1       auto    on      trunking    TE          2       port-channel 21
fc2/10      1       auto    on      trunking    TE          2       port-channel 21
...
fc2/13      999     auto    on      up          F           1       --
fc2/14      999     auto    on      up          FL          1       --
fc2/15      1       SD      --      up          SD          2       --
fc2/16      1       auto    on      trunking    TE          2       --
```

```
--------------------------------------------------------------------------------
Interface           Status      Speed
                                (Gbps)
--------------------------------------------------------------------------------
sup-fc0             up          1

--------------------------------------------------------------------------------
Interface           Status      IP Address              Speed       MTU
--------------------------------------------------------------------------------
mgmt0               up          172.22.36.175/22        100 Mbps    1500

--------------------------------------------------------------------------------
Interface           Status      IP Address              Speed       MTU--
--------------------------------------------------------------------------------
vsan5               up          10.10.10.1/24           1 Gbps      1500
--------------------------------------------------------------------------------
Interface           Vsan        Admin           Status      Oper    Oper
                                Trunk                       Mode    Speed
                                Mode                                (Gbps)
--------------------------------------------------------------------------------
port-channel 21     1           on              trunking    TE      4
--------------------------------------------------------------------------------
Interface           Status      Dest IP Addr    Src IP Addr     TID    Explicit Path
--------------------------------------------------------------------------------
fc-tunnel 100       up          10.10.10.2      10.10.10.1      100
```

Example 11-11 Displays Detailed Information for the ST Port Interface

```
switch# show interface fc1/11
fc1/11 is up
    Hardware is Fibre Channel
    Port WWN is 20:0b:00:05:30:00:59:de
    Admin port mode is ST
    Port mode is ST
    Port vsan is 1
    Speed is 1 Gbps
    Rspan tunnel is fc-tunnel 100
    Beacon is turned off
    5 minutes input rate 248 bits/sec, 31 bytes/sec, 0 frames/sec
    5 minutes output rate 176 bits/sec, 22 bytes/sec, 0 frames/sec
      6862 frames input, 444232 bytes
        0 discards, 0 errors
        0 CRC, 0 unknown class
        0 too long, 0 too short
      6862 frames output, 307072 bytes
        0 discards, 0 errors
      0 input OLS, 0 LRR, 0 NOS, 0 loop inits
      0 output OLS, 0 LRR, 0 NOS, 0 loop inits
```

Example 11-12 Displays the FC Tunnel Status

```
switch# show fc-tunnel
fc-tunnel is enabled
```

Example 11-13 Displays FC Tunnel Egress Mapping Information

```
switch# show fc-tunnel tunnel-id-map
tunnel id egress interface
    150     fc3/1
    100     fc3/1
```

Note Multiple tunnel IDs can terminate at the same interface.

Example 11-14 Displays FC Tunnel Explicit Mapping Information

```
switch# show fc-tunnel explicit-path
Explicit path name: Alternate1
       10.20.1.2 loose
       10.20.1.3 strict
Explicit path name: User2
       10.20.50.1 strict
       10.20.50.4 loose
```

Example 11-15 Displays SPAN Mapping Information

```
switch# show span session
Session 2 (active)
   Destination is fc-tunnel 100
   No session filters configured
   Ingress (rx) sources are
     fc2/16,
   Egress (tx) sources are
     fc2/16,
```

Example 11-16 Displays the FC Tunnel Interface

```
switch# show interface fc-tunnel 200
fc-tunnel 200 is up
Dest   IP Addr: 200.200.200.7   Tunnel ID: 200
Source IP Addr: 200.200.200.4   LSP ID: 1
Explicit Path Name:
```

Default SPAN and RSPAN Settings

Table 11-1 lists the default settings for SPAN parameters.

Table 11-1 Default SPAN Configuration Parameters

Parameters	Default
SPAN session	Active.
If filters are not specified	SPAN traffic includes traffic through a specific interface from all active VSANs.
Encapsulation	Disabled.
SD port	Output frame format is Fibre Channel.

Table 11-2 lists the default settings for RSPAN parameters.

Table 11-2 Default RSPAN Configuration Parameters

Parameters	Default
FC tunnel	Disabled
Explicit path	Not configured
Minimum cost path	Used if explicit path is not configured

CHAPTER 12

Configuring Fabric Configuration Server

This chapter describes the Fabric Configuration Server (FCS) feature provided in the Cisco MDS 9000 Family of directors and switches. It includes the following sections:

About FCS

The Fabric Configuration Server (FCS) provides discovery of topology attributes and maintains a repository of configuration information of fabric elements. A management application is usually connected to the FCS on the switch through an N port. The FCS views the entire fabric based on the following objects:

- Interconnect element (IE) object—Each switch in the fabric corresponds to an IE object. One or more IE objects form a fabric.
- Port object—Each physical port in an IE corresponds to a port object. This includes the switch ports (xE, Fx, and TL ports) and their attached Nx ports.
- Platform object—A set of nodes may be defined as a platform object to make it a single manageable entity. These nodes are end-devices (host systems, storage subsystems) attached to the fabric. Platform objects reside at the edge switches of the fabric.

Each object has its own set of attributes and values. A null value may also be defined for some attributes.

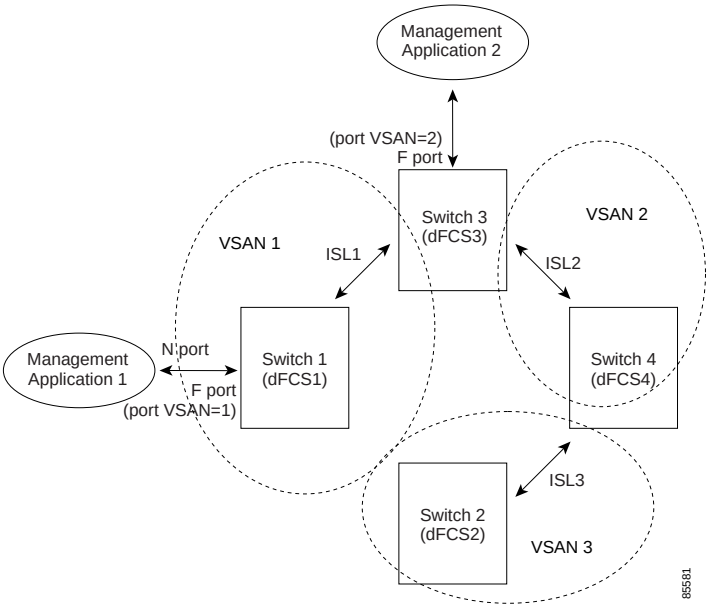
In the Cisco MDS 9000 Family switch environment, multiple VSANs constitute a fabric, where one instance of the FCS is present per VSAN.

As of Cisco NX-OS Release 4.1(1), FCS supports the discovery of virtual devices. The **fcs virtual-device-add** command, issued in FCS configuration submode, allows you to discover virtual devices in a particular VSAN or in all VSANs. The devices that are zoned for IVR must be discovered with this command and have request domain_ID (RDI) enabled, before activating the IVR zone set.

If you have attached a management application to a switch, all the frames directed towards the FCS in the switch are part of the port VSAN in the switch port (Fx port). Your view of the management application is limited only to this VSAN. However, information about other VSANs that this switch is part of can be obtained either through the SNMP or CLI.

In Figure 12-1 Management Application 1 (M1) is connected through an F port with port VSAN ID 1, and Management Application 2 (M2) is connected through an F port with port VSAN ID 2. M1 can query the FCS information of switches S1 and S3, and M2 can query switches S3 and S4. Switch S2 information is not known to both of them. FCS operations can be done only on those switches that are visible in the VSAN. Note that M2 can send FCS requests only for VSAN 2 even though S3 is also a part of VSAN 1.

Figure 12-1 FCSs in a VSAN Environment



Significance of FCS

This section lists the significance of FCSs.

- FCSs support network management including the following:
  - N port management application can query and obtain information about fabric elements.
  - SNMP manager can use the FCS management information base (MIB) to start discovery and obtain information about the fabric topology.
- FCSs support TE and TL ports in addition to the standard F and E ports.
- FCS can maintain a group of modes with a logical name and management address when a platform registers with it. FCSs maintain a backup of all registrations in secondary storage and update it with every change. When a restart or switchover happens, FCSs retrieve the secondary storage information and rebuild its database.
- SNMP manager can query FCSs for all IEs, ports, and platforms in the fabric.

FCS Name Specification

You can specify if the unique name verification is for the entire fabric (globally) or only for locally (default) registered platforms.

Note Set this command globally only if all switches in the fabric belong to the Cisco MDS 9000 Family.

To enable global checking of the platform name, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcs plat-check-global vsan 1`	Enables global checking of the platform name.
	`switch(config)# no fcs plat-check-global vsan 1`	Disables (default) global checking of the platform name.

To register platform attributes, follow these steps:

	Command	Purpose
Step 1	`switch# config t` `switch(config)#`	Enters configuration mode.
Step 2	`switch(config)# fcs register` `switch(config-fcs-register)#`	Enters the FCS registration submode.
Step 3	`switch(config-fcs-register)# platform name SamplePlatform vsan 1` `switch(config-fcs-register-attrib)#`	Enters the FCS registration attributes submode.
	`switch(config-fcs-register)# no platform name SamplePlatform vsan 1` `switch(config-fcs-register)#`	Deletes a registered platform.
Step 4	`switch(config-fcs-register-attrib)# mgmt-addr 1.1.1.1`	Configures the platform management IPv4 address.
	`switch(config-fcs-register-attrib)# no mgmt-addr 1.1.1.1`	Deletes the platform management IPv4 address.
	`switch(config-fcs-register-attrib)# mgmt-addr 2001:0DB8:800:200C::417A`	Configures the platform management IPv6 address.
	`switch(config-fcs-register-attrib)# no mgmt-addr 2001:0DB8:800:200C::417A`	Deletes the platform management IPv6 address.
Step 5	`switch(config-fcs-register-attrib)# nwwn 11:22:33:44:55:66:77:88`	Configures the platform node name.
	`switch(config-fcs-register-attrib)# no nwwn 11:22:33:44:55:66:77:88`	Deletes the platform node name.
Step 6	`switch(config-fcs-register-attrib)# type 5`	Configures the fc-gs-3 defined platform type.
	`switch(config-fcs-register-attrib)# no type 5`	Deletes the configured type and reverts the switch to its factory default of unknown type.
Step 7	`switch(config-fcs-register-attrib)# exit`	Exits the FCS registration attributes submode.
Step 8	`switch(config-fcs-register)# exit` `switch(config)#`	Exits the FCS registration submode.

Displaying FCS Elements

Use the **show fcs** commands to display the status of the WWN configuration (see Example 12-1 to 12-9).

Example 12-1 Displays FCS Local Database Information

```
switch# show fcs database
FCS Local Database in VSAN: 1
------------------------------
Switch WWN               : 20:01:00:05:30:00:16:df
Switch Domain Id         : 0x7f(127)
Switch Mgmt-Addresses    : snmp://172.22.92.58/eth-ip
                           http://172.22.92.58/eth-ip
Fabric-Name              : 20:01:00:05:30:00:16:df
Switch Logical-Name      : 172.22.92.58
Switch Information List  : [Cisco Systems*DS-C9509*0*20:00:00:05:30:00
Switch Ports:
-------------------------------------------------------------------
Interface  pWWN                      Type     Attached-pWWNs
-------------------------------------------------------------------
fc2/1      20:41:00:05:30:00:16:de  TE       20:01:00:05:30:00:20:de
fc2/2      20:42:00:05:30:00:16:de  Unknown  None
fc2/17     20:51:00:05:30:00:16:de  TE       20:0a:00:05:30:00:20:de

FCS Local Database in VSAN: 5
------------------------------
Switch WWN               : 20:05:00:05:30:00:12:5f
Switch Domain Id         : 0xef(239)
Switch Mgmt-Addresses    : http://172.22.90.171/eth-ip
                           snmp://172.22.90.171/eth-ip
                           http://10.10.15.10/vsan-ip
                           snmp://10.10.15.10/vsan-ip
Fabric-Name              : 20:05:00:05:30:00:12:5f
Switch Logical-Name      : 172.22.90.171
Switch Information List  : [Cisco Systems*DS-C9509**20:00:00:05:30:00:12:5e]
Switch Ports:
-------------------------------------------------------------------
Interface  pWWN                      Type     Attached-pWWNs
-------------------------------------------------------------------
fc3/1      20:81:00:05:30:00:12:5e  TE       22:01:00:05:30:00:12:9e
fc3/2      20:82:00:05:30:00:12:5e  TE       22:02:00:05:30:00:12:9e
fc3/3      20:83:00:05:30:00:12:5e  TE       22:03:00:05:30:00:12:9e
```

Example 12-2 Displays a List of All IEs for a Specific VSAN

```
switch# show fcs ie vsan 1
IE List for VSAN: 1
-------------------------------------------------------------------
IE-WWN                  IE-Type                          Mgmt-Id
-------------------------------------------------------------------
20:01:00:05:30:00:16:df  Switch (Local)                   0xfffc7f
20:01:00:05:30:00:20:df  Switch (Adjacent)                0xfffc64
[Total 2 IEs in Fabric]
```

Example 12-3 Displays Interconnect Element Object Information for a Specific nWWN

```
switch# show fcs ie nwwn 20:01:00:05:30:00:16:df vsan 1
IE Attributes
-------------
Domain-Id = 0x7f(127)
Management-Id = 0xfffc7f
Fabric-Name = 20:01:00:05:30:00:16:df
Logical-Name = 172.22.92.58
Management Address List =
        snmp://172.22.92.58/eth-ip
        http://172.22.92.58/eth-ip
Information List:
        Vendor-Name = Cisco Systems
        Model Name/Number = DS-C9509
        Release-Code = 0
```

Example 12-4 Displays Information for a Specific Platform

```
switch# show fcs platform name SamplePlatform vsan 1
Platform Attributes
-------------------
Platform Node Names:
        11:22:33:44:55:66:77:88
Platform Type = Gateway
Platform Management Addresses:
        1.1.1.1
```

Example 12-5 Displays a List of Platforms for a Specified VSAN

```
switch# show fcs platform vsan 1
Platform List for VSAN: 1
Platform-Names
--------------
SamplePlatform
[Total 1 Platforms in Fabric]
```

Example 12-6 Displays a List of Switch Ports in a Specified VSAN

```
switch# show fcs port vsan 24
Port List in VSAN: 24
                -- IE WWN: 20:18:00:05:30:00:16:df  --
------------------------------------------------------------------------
Port-WWN                Type      Module-Type            Tx-Type
------------------------------------------------------------------------
20:41:00:05:30:00:16:de  TE_Port   SFP with Serial Id      Shortwave Laser
20:51:00:05:30:00:16:de  TE_Port   SFP with Serial Id      Shortwave Laser
[Total 2 switch-ports in IE]
                -- IE WWN: 20:18:00:05:30:00:20:df  --
------------------------------------------------------------------------
Port-WWN                Type      Module-Type            Tx-Type
------------------------------------------------------------------------
20:01:00:05:30:00:20:de  TE_Port   SFP with Serial Id      Shortwave Laser
20:0a:00:05:30:00:20:de  TE_Port   SFP with Serial Id      Shortwave Laser
[Total 2 switch-ports in IE]
```

Example 12-7 Displays Port Information for a Specified pWWN

```
switch# show fcs port pwwn 20:51:00:05:30:00:16:de vsan 24
Port Attributes
---------------
Port Type = TE_Port
Port Number = 0x1090000
Attached-Port-WWNs:
        20:0a:00:05:30:00:20:de
Port State = Online
```

Example 12-8 Displays FCS Statistics

```
switch# show fcs statistics
FCS Statistics for VSAN: 1
--------------------------
FCS Rx Get Reqs   :2
FCS Tx Get Reqs   :7
FCS Rx Reg Reqs   :0
FCS Tx Reg Reqs   :0
FCS Rx Dereg Reqs :0
FCS Tx Dereg Reqs :0
FCS Rx RSCNs      :0
...
FCS Statistics for VSAN: 30
--------------------------
FCS Rx Get Reqs   :2
FCS Tx Get Reqs   :2
FCS Rx Reg Reqs   :0
FCS Tx Reg Reqs   :0
FCS Rx Dereg Reqs :0
FCS Tx Dereg Reqs :0
FCS Rx RSCNs      :0
FCS Tx RSCNs      :0
...
```

Example 12-9 Displays Platform Settings for Each VSAN

```
switch# show fcs vsan
------------------------------
VSAN    Plat Check fabric-wide
------------------------------
0001    Yes
0010    No
0020    No
0021    No
0030    No
```

Default Settings

Table 12-1 lists the default FCS settings.

Table 12-1 Default FCS Settings

Parameters	Default
Global checking of the platform name	Disabled.
Platform node type	Unknown.

INDEX

Numerics

A

B

C

D

E

F

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

Z